

**LAPORAN PRAKTIKUM**  
**POSTTEST 6**  
**ALGORITMA PEMROGRAMAN DASAR**

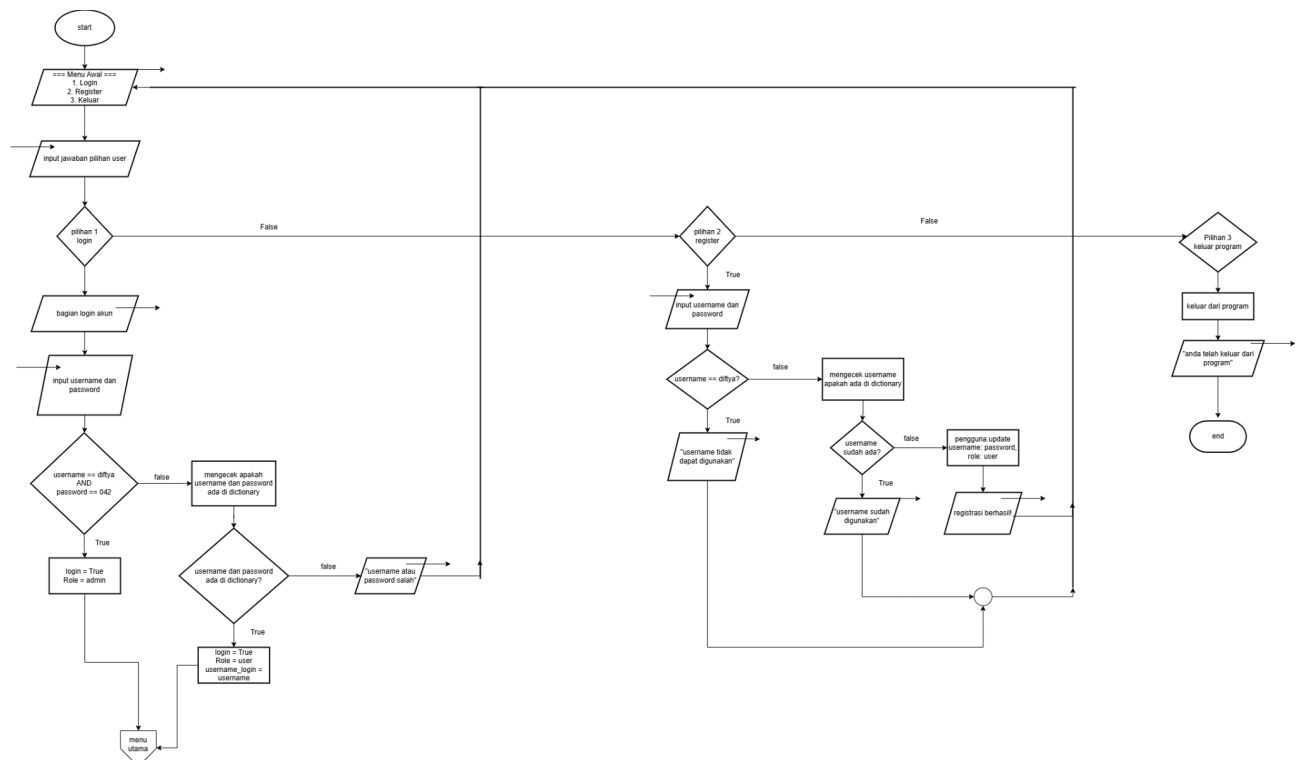


**Disusun oleh:**  
**Diftya Azzahra (2509106042)**  
**Kelas (A2'25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

## 1. Flowchart

Berikut adalah flowchart yang saya buat



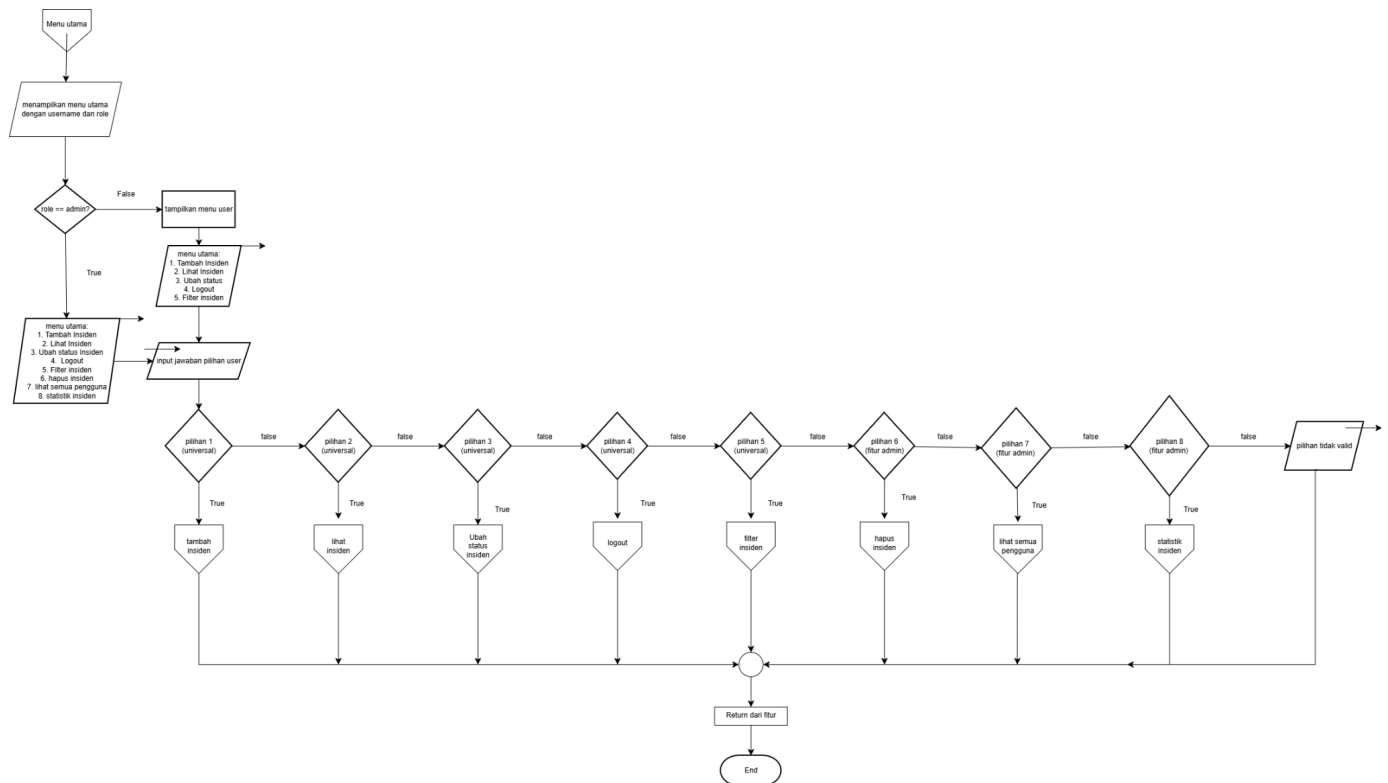
Gambar 1.1 Menu awal

1. **Start (Main):** Program dimulai
2. **Output menampilkan menu awal:** Di menu awal, sistem akan menampilkan menu yang menampilkan tiga pilihan yaitu Login, Register, dan Keluar.
3. **Input jawaban user:** Setelah menu ditampilkan, sistem **meminta input nomor** pilihan menu dari user (1 sampai 3).
4. **Decision:** Pada bagian ini, program akan menerima input berupa pilihan menu. Jika user memilih **1** maka akan masuk ke fitur login, jika memilih **2** maka akan masuk ke fitur register, jika memilih menu ke **3** maka user akan keluar dari program. Jika user memilih pilihan di luar angka dan pilihan yang disediakan, sistem akan menampilkan pesan "**menu tidak valid**".
5. **Alur Pilihan 1 - Login**
6. Jika user memilih login, sistem akan masuk ke **pilihan 1 (login)**. Jika **true**, program akan menampilkan **bagian login** dan kemudian meminta **input username dan password**. Setelah data diinput, sistem melakukan pengecekan **username == diftya AND password == 042** untuk memvalidasi apakah penginput merupakan admin. Jika **true**, maka sistem akan mengatur **login = True dan Role = admin** kemudian

mengarah ke **menu utama**. Jika **false**, sistem akan melakukan pengecekan kedua yaitu username dan password ada di dictionary atau tidak. Untuk mengecek apakah user tersebut terdaftar sebagai user biasa.

7. Kalau masih tidak ada, muncul pesan "username atau password salah" lalu balik ke menu awal. Jika ada, login berhasil sebagai user biasa dan masuk ke menu utama.

## 8. Alur Pilihan 2 - Register

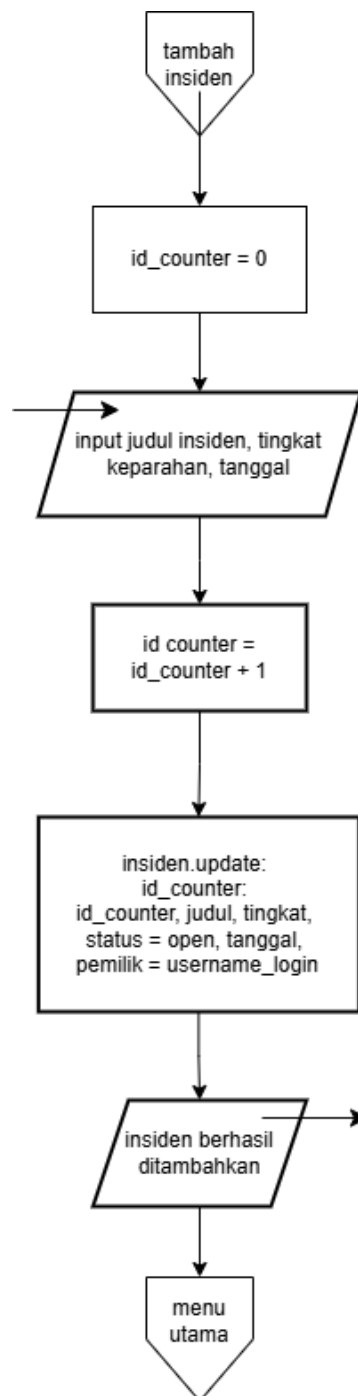


Gambar 1.2 Menu Utama

9. Dari menu utama user memasukkan pilihan 2 (Register), maka flowchart akan masuk ke cabang pilihan.
10. Program akan meminta input berupa username dan password. Lalu sistem akan mengecek apakah username sama dengan admin. Hal ini dibuat agar user biasa tidak dapat memakai username admin. Jika **true** maka akan menampilkan pesan username tidak dapat dipakai.
11. Jika **false** maka sistem akan mengecek apakah username yang diinput oleh user terdapat pada dictionary **pengguna**. Jika **true** maka akan memunculkan pesan username telah terdaftar. Namun jika **false**, maka username artinya username belum dipakai dan pengguna berhasil mendaftar.

## 12. Alur pilihan 3 - Keluar

13. Jika pengguna memilih pilihan 3 pada menu utama, maka program akan mengeluarkan output berupa pesan “**Anda telah keluar dari program**”. Lalu **break** (tidak terlihat pada flowchart) akan dijalankan agar pengguna otomatis keluar dari program.
14. Setelah login berhasil, user masuk ke **menu utama** (isi menu berbeda-beda) tergantung role.
15. Di menu utama, menampilkan daftar pilihan berbeda sesuai role. User biasa dapat 5 menu: Tambah Insiden, Lihat Insiden, Ubah Status, Logout, dan Filter Insiden. Admin dapat 8 menu, ditambah 3 menu khusus yaitu Hapus Insiden, Lihat Semua Pengguna, dan Statistik Status Insiden. User diminta **menginput** pilihan menu.
16. **Pilihan 1** kalau true masuk **Tambah insiden**. **Pilihan 2** kalau true masuk **Lihat insiden**. **Pilihan 3** kalau true masuk **Ubah status insiden**. **Pilihan 4** kalau true masuk **logout**. **Pilihan 5** kalau true masuk **filter insiden**. **Pilihan 6** (admin) kalau true masuk **hapus insiden**. **Pilihan 7** (admin) kalau true masuk **Lihat semua pengguna**. **Pilihan 8** kalau true masuk **Ubah insiden**.
17. Jika semua pilihan false, berarti user input nomor yang salah, jadi masuk ke **pilihan tidak valid**.



Gambar 1.3 Fitur tambah insiden

18. Dijalankan saat pengguna memilih fitur “tambah insiden”. pertama-tama dilakukan inisialisasi pada variabel **id\_counter = 0**. Ini dilakukan agar program menghitung dari 1 (jika ada insiden baru yang masuk, id nya akan langsung 1 bukan 0).
19. Sistem akan meminta pengguna memasukkan data insiden seperti judul kejadian, tingkat keparahan, dan tanggal kejadian. Setelah semua input diterima, program membuat entri baru dengan menambahkan data ke daftar insiden menggunakan

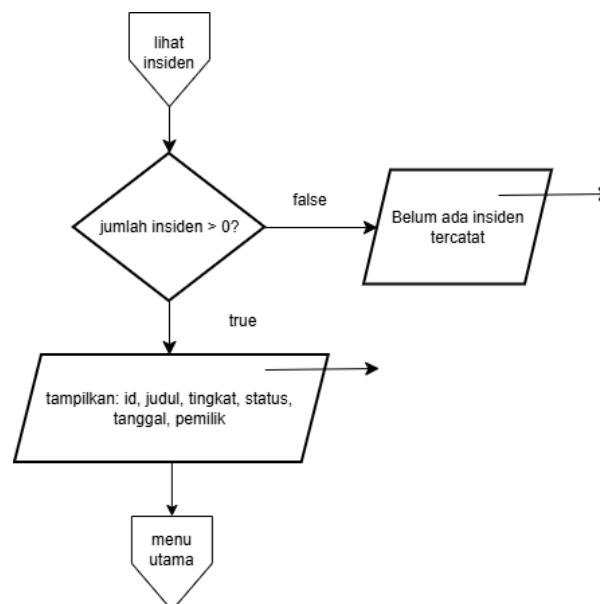
```

id_counter += 1
insiden.update({
    id_counter: {
        "id_counter" : id_counter,
        "judul" : judul,
        "tingkat" : tingkat,
        "status" : "open",
        "tanggal" : tanggal,
        "pemilik": username_login,
    }
})

```

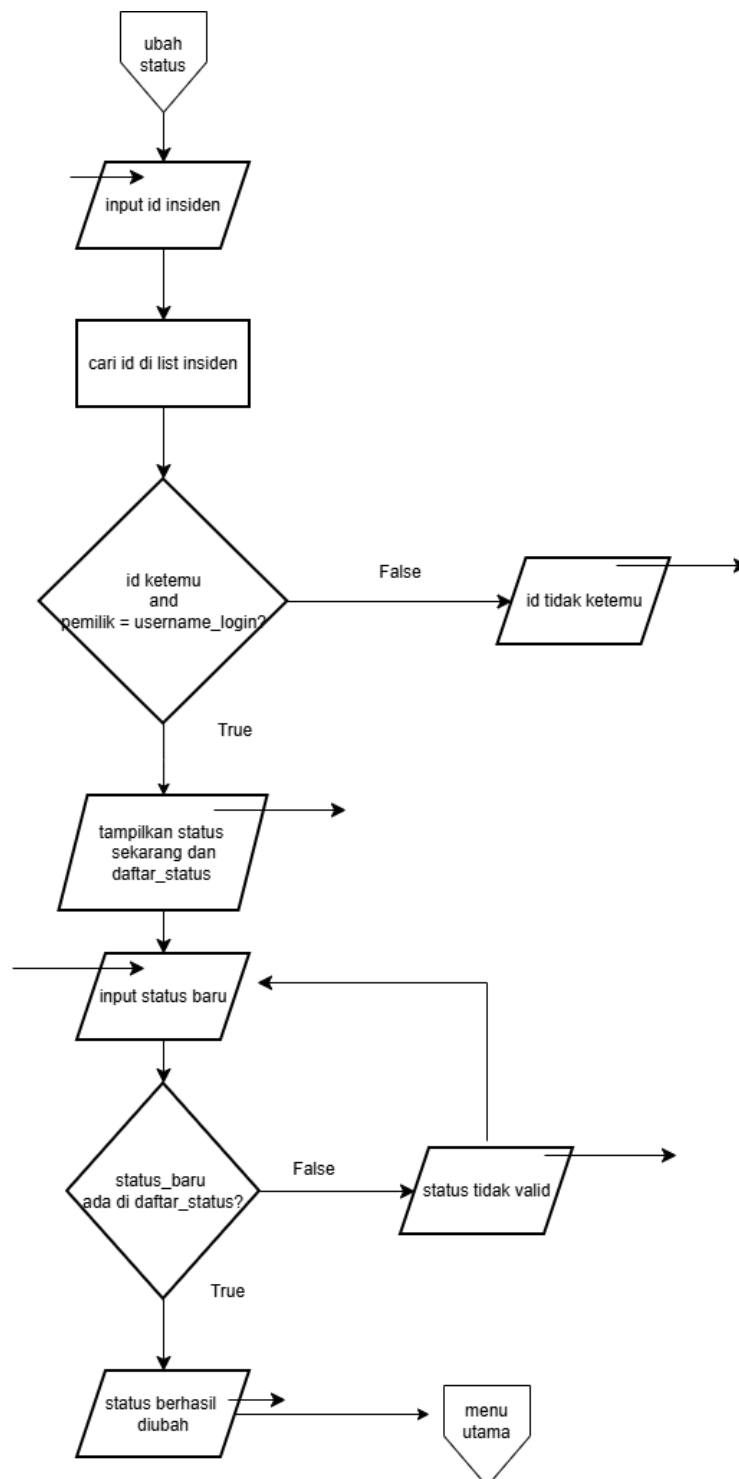
*Note:* Saya sengaja mencantumkan blok kode dictionary agar tidak terjadi miskonsepsi pada aslab yang pada akhirnya akan menyangkut flowchart. **id\_counter +1** adalah penghitung id. Setiap kali pengguna menambahkan insiden baru, nilainya bertambah 1. Untuk **insiden.update**, insiden merupakan dictionary utama yang berisi semua data insiden. lalu **.update** merupakan cara untuk menambahkan data baru ke dictionary yang pada posttest sebelumnya merupakan .append karna tipe data masih berupa list. **id\_counter** di dalam kurung kurawal pertama merupakan key utama di dictionary insiden. Nantinya, pada menu-menu lain pun akan menggunakan ID untuk mengidentifikasi sebuah insiden seperti pada fitur menghapus insiden.

20. Ketika insiden berhasil ditambahkan. Maka sistem akan memberikan output berupa pesan “**Insiden berhasil ditambahkan**”.



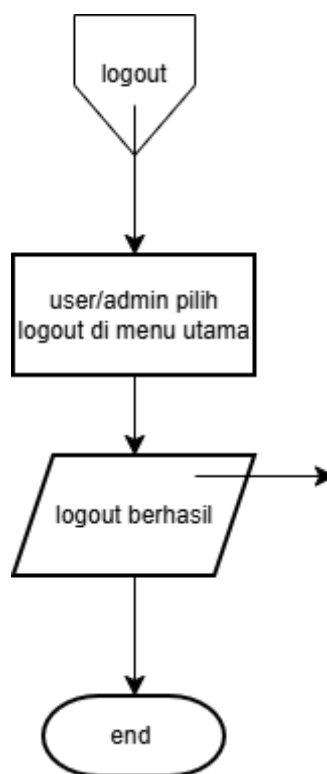
Gambar 1.4 Fitur lihat insiden

21. Proses dimulai dengan pengecekan apakah ada insiden yang tersimpan di sistem. Kalau belum ada sistem akan langsung menampilkan pesan **“Belum ada insiden tercatat”**. Jika ada, sistem akan langsung menampilkan id insiden, judul, tingkat, status, tanggal terjadinya, dan pemilik atau penginput dari insiden tersebut.



Gambar 1.5 Ubah status insiden

22. Dimulai ketika pengguna ingin **mengubah status insiden** dengan memilih menu nomor **3**. Pengguna perlu memasukkan **ID insiden** yang ingin diubah. Setelah itu, sistem mencari ID tersebut di daftar insiden. Kalau ID tidak ditemukan atau bukan milik pengguna yang sedang login, sistem akan menampilkan pesan **“ID tidak ketemu”** dan proses berhenti. Namun jika ID ditemukan sistem akan menampilkan **status insiden saat ini**, lalu meminta pengguna untuk memasukkan **status baru**. Sebelum pengguna memasukkan status baru, program terlebih dahulu memberikan pilihan status yaitu dengan memunculkan variabel **daftar\_status** yang berisi pilihan status untuk dipilih. Setelah status baru dimasukkan, sistem memeriksa apakah status tersebut valid (tersedia di daftar\_status). Jika status yang dimasukkan tidak valid, maka muncul pesan **“status tidak valid”** dan meminta user untuk memasukkan ulang status. Tapi jika statusnya valid, sistem akan mengganti status lama dengan status baru dan menampilkan pesan **“status berhasil diubah”**.

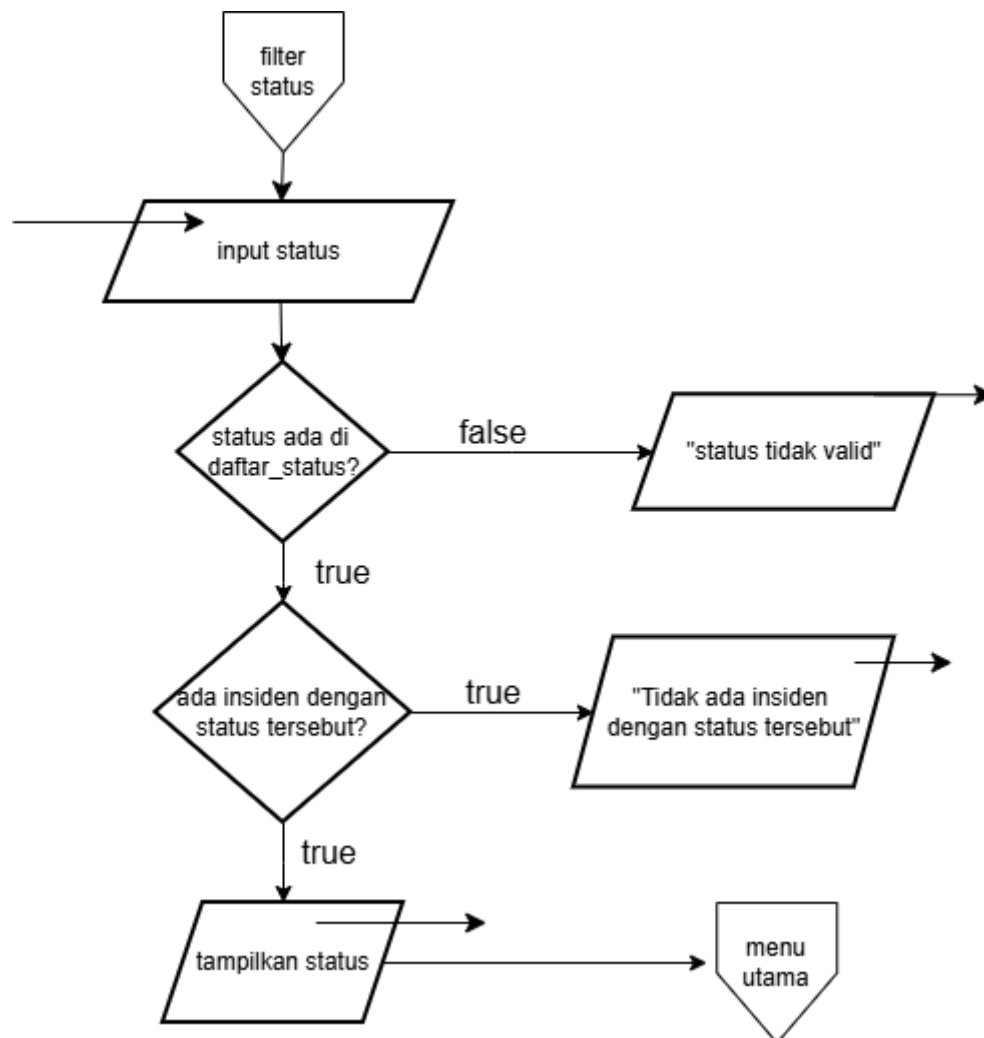


Gambar 1.6 Fitur Logout

23. Berjalan ketika user memilih menu logout. Proses dimulai dari menu utama ketika user atau admin memilih opsi **logout**. Setelah pilihan itu dipilih, sistem langsung menjalankan proses logout dan menampilkan pesan **“Logout berhasil”**. Setelah itu,

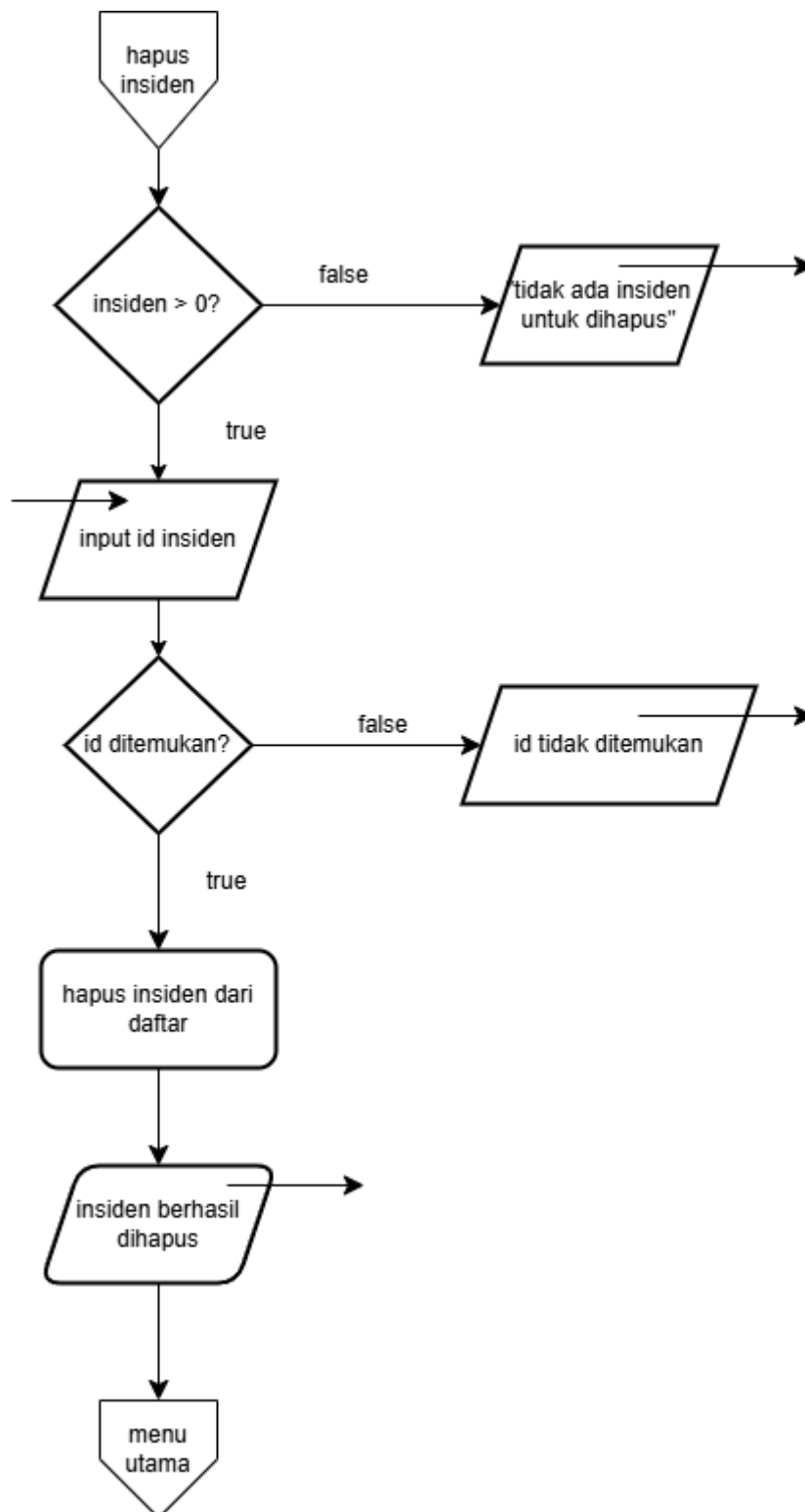


alur berakhir di **end**, yang menandakan sesi pengguna sudah selesai dan mereka tidak lagi terhubung ke sistem.



Gambar 1.7 Fitur filter status insiden

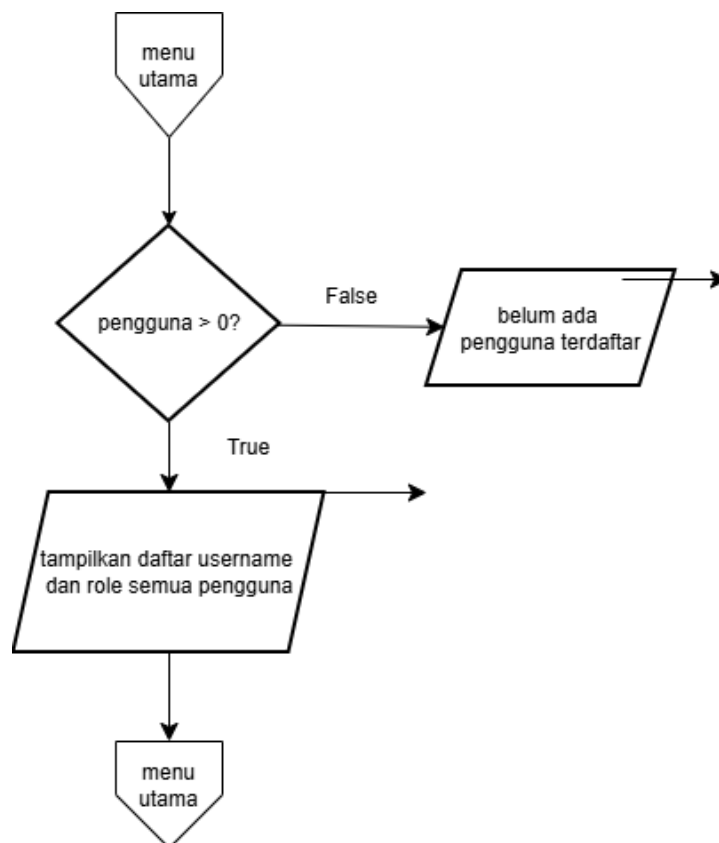
24. Setelah itu, masuk ke bagian untuk proses saat pengguna ingin memfilter daftar insiden berdasarkan status tertentu. Proses dimulai ketika user memilih nomor 5 di menu **utama**, lalu pengguna diminta untuk **memasukkan status yang ingin difilter**. Setelah itu, sistem akan memeriksa apakah status tersebut valid (tersedia di daftar\_status). Jika statusnya tidak valid, sistem memberikan output berupa pesan “status tidak valid” dan meminta user untuk memasukkan ulang status. Namun, jika statusnya valid, sistem akan **menampilkan daftar insiden** yang sesuai dengan status tersebut.



Gambar 1.8 Fitur menghapus insiden (Khusus admin)

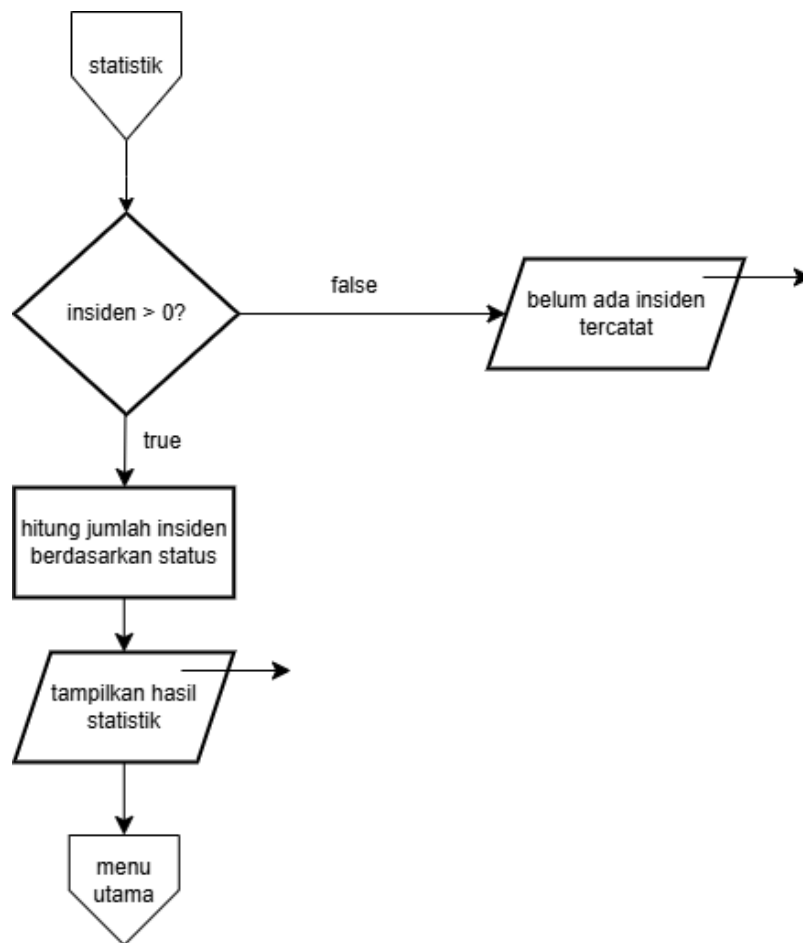
25. Bagian ini menjelaskan proses penghapusan insiden dari sistem. Proses dimulai saat admin memilih menu **hapus insiden**. Sistem terlebih dahulu mengecek apakah ada insiden yang tersimpan. Jika tidak ada, sistem langsung menampilkan pesan “**tidak**

**ada insiden untuk dihapus**". Namun, jika ada, sistem meminta admin untuk memasukkan **ID insiden** yang ingin dihapus. Setelah itu, sistem mencari ID tersebut dalam daftar insiden. Jika ID tidak ditemukan, muncul pesan "**ID tidak ditemukan**". Tapi jika ID nya ditemukan, sistem menghapus insiden dari daftar dan menampilkan pesan "**insiden berhasil dihapus**". Setelah proses selesai, alur kembali ke **menu utama**.



Gambar 1.9 Fitur Lihat Semua Pengguna (Khusus admin)

26. Dimulai ketika admin memilih menu **lihat semua pengguna**. Setelah fitur ini dijalankan, sistem akan memeriksa apakah ada pengguna yang sudah terdaftar di dalam sistem. Jika tidak ada, maka sistem menampilkan pesan "**belum ada pengguna terdaftar**". Namun, jika ada pengguna yang tersimpan, sistem akan **menampilkan daftar username beserta role** dari semua pengguna yang ada.



Gambar 1.10 Fitur Statistik (khusus admin)

27. Dimulai saat admin memilih menu **statistik** untuk melihat data insiden. Setelah fitur dijalankan, sistem akan memeriksa apakah ada insiden yang tersimpan di dalam data. Jika tidak ada, sistem langsung menampilkan pesan “**belum ada insiden tercatat**”. Namun, jika ada insiden tercatat, sistem akan **menghitung jumlah insiden berdasarkan statusnya**, misalnya berapa yang masih terbuka, sedang diproses, atau sudah selesai. Setelah perhitungan selesai, sistem akan menampilkan hasil statistik tersebut, lalu alur kembali ke menu utama.

## 2. Deskripsi Singkat Program

Program ini diibaratkan sebagai aplikasi di sebuah perusahaan keamanan IT. Jika terjadi suatu insiden seperti kebocoran data atau teretasnya suatu sistem maka pekerja biasa bisa menginput kejadian yang terjadi agar bisa dievaluasi lebih lanjut. Lalu admin adalah manager atau Team Lead yang dapat mengawasi serta memberi tahu apa saja yang harus

dilakukan. Username mewakili nama masing-masing pekerja. Username dibuat tidak boleh sama agar team lead tidak kebingungan pekerja mana yang menginput atau menemukan insiden tersebut.

### 3. Source Code

```
import os

print("=== Incident Respond Management System Program ===")

# Data insiden
insiden = {}

# Data admin dan pengguna
admin = ("diftya", "042")
pengguna = {}

id_counter = 0
login = False
username_login = ""
role = ""

# Daftar status
daftar_status = {
    "open" : "Insiden baru dilaporkan",
    "in progress" : "Insiden sedang ditangani",
    "resolved" : "Insiden berhasil diselesaikan",
    "closed" : "Kasus telah ditutup",
}

while True:
    print("=== SISTEM MANAJEMEN INSIDEN (SIEM) ===")
    print("1. Login")
    print("2. Register")
    print("3. Keluar")
    menu_awal = input("Pilih menu: ")

    if menu_awal == "1":
        os.system('cls')
        print("===== LOGIN =====")
        username = input("Username: ")
        password = input("Password: ")

        if username == admin[0] and password == admin[1]:
            login = True
            username_login = username
            role = "admin"
```

```

        print("Login Berhasil")
    else:
        ketemu = False
        for user, data in pengguna.items():
            if user == username and data["password"] == password:
                login = True
                username_login = username
                role = data["role"]
                ketemu = True
                break
        if not ketemu:
            print("Username atau passwordmu salah")
            input("Tekan Enter untuk melanjutkan program")
os.system('cls')

# Fitur Register
elif menu_awal == "2":
    os.system('cls')
    print("===== REGISTER =====")
    username = input("Masukkan username anda: ")
    password = input("Masukkan password: ")
    username_sama = False
    if username == admin[0]:
        print("Username ini tidak bisa dipakai.")
    else:
        for user, data in pengguna.items():
            if user == username:
                username_sama = True
                break
        if username_sama:
            print("Username telah terdaftar.")
        else:
            pengguna.update({
                username: {
                    "password": password,
                    "role": "user"
                }
            })

            print("Selamat registrasi berhasil")
            input("Tekan Enter untuk lanjut ke menu")
            os.system('cls')

# Fitur Keluar Program
elif menu_awal == "3":
    print("Anda telah keluar dari program")
    break

```

```

else:
    print("Menu tidak valid")
    input("Tekan Enter untuk lanjut")
    os.system('cls')
# Setelah Login
while login:
    os.system('cls')
    if role == "user":
        print("===== MENU UTAMA =====")
        print(f"Login sebagai: {username_login} ({role})")
        print("1. Tambah Insiden")
        print("2. Lihat Insiden")
        print("3. Ubah Status Insiden")
        print("4. Logout")
        print("5. Filter Insiden berdasarkan Status")

    elif role == "admin":
        print("===== MENU UTAMA (ADMIN) =====")
        print(f"Login sebagai: {username_login} ({role})")
        print("1. Tambah Insiden")
        print("2. Lihat Insiden")
        print("3. Ubah Status Insiden")
        print("4. Logout")
        print("5. Filter Insiden berdasarkan Status")
        print("6. Hapus Insiden")
        print("7. Lihat Semua Pengguna")
        print("8. Statistik Insiden")

    pilih_menu = input("Pilih menu: ")

    # Tambah Insiden
    if pilih_menu == "1":
        os.system('cls')
        print("===== Tambah Insiden =====")
        judul = input("Judul insiden: ")
        tingkat = input("Tingkat keparahan: ")
        status = "open"
        tanggal = input("Tanggal (DD/MM/YYYY): ")

        id_counter += 1
        insiden.update({
            id_counter: {
                "id_counter" : id_counter,
                "judul" : judul,
                "tingkat" : tingkat,
                "status" : "open",
                "tanggal" : tanggal,

```

```

        "pemilik": username_login,
    }
})
print("Insiden berhasil ditambahkan!")
input("Tekan Enter untuk melanjutkan")

# Lihat Insiden
elif pilih_menu == "2":
    os.system('cls')
    print("===== Lihat Daftar Insiden
=====")
    if len(insiden) == 0:
        print("Belum ada insiden tercatat.")
    else:
        for id_insiden, data in insiden.items():
            if role == "admin" or data["pemilik"] == username_login:
                print(f"ID: {data['id_counter']}, Judul:
{data['judul']}, "
                    f" Tingkat: {data['tingkat']}, "
                    f"Status: {data['status']}, Tanggal:
{data['tanggal']}, Pemilik: {data['pemilik']}")
                input("Tekan Enter untuk melanjutkan program")

# Ubah Status Insiden
elif pilih_menu == "3":
    os.system('cls')
    print("===== Mengubah Status Insiden =====")
    ubah_id = input("Masukkan ID insiden: ")
    ketemu = False

    for id_insiden, data in insiden.items():
        if str(data["id_counter"]) == ubah_id and data["pemilik"] ==
username_login:
            print(f"Status sekarang: {data['status']}")
            print("Pilihan status:", daftar_status)
            status_baru = input("Masukkan status baru: ")
            while status_baru not in daftar_status:
                print("Status tidak valid! Pilihan:", daftar_status)
                status_baru = input("Masukkan status lagi: ")

            if status_baru in daftar_status:
                data["status"] = status_baru
                print("Status berhasil diubah!")
            else:
                print("Status tidak valid!")
            ketemu = True
            break
    if not ketemu:

```



```

        print("ID tidak ketemu atau bukan milik Anda.")
        input("Tekan Enter untuk melanjutkan")

# Logout
elif pilih_menu == "4":
    login = False
    username_login = ""
    role = ""
    print("Logout berhasil.")
    input("Tekan Enter untuk kembali ke menu awal")

# Filter Insiden
elif pilih_menu == "5":
    os.system('cls')
    print("===== Filter Status Insiden
=====")
    status_filter = input("Masukkan status untuk filter: ")
    while status_filter not in daftar_status:
        print("Status tidak valid! Pilihan:", daftar_status)
        status_filter = input("Masukkan status lagi: ")
    ditemukan = False
    for id_insiden, data in insiden.items():
        if data["status"] == status_filter and (role == "admin" or
data["pemilik"] == username_login):
            print(f"ID: {data['id_counter']}, Judul:
{data['judul']}, Status: {data['status']}, Pemilik: {data['pemilik']}")
            ditemukan = True
    if not ditemukan:
        print("Tidak ada insiden dengan status tersebut.")
    input("Tekan Enter untuk lanjut")

# Hapus Insiden (Admin)
elif role == "admin" and pilih_menu == "6":
    os.system('cls')
    print("===== Hapus Insiden
=====")
    if len(insiden) == 0:
        print("Tidak ada insiden untuk dihapus.")
    else:
        hapus_id = input("Masukkan ID insiden yang ingin dihapus: ")
        ketemu = False
        for id_insiden, data in insiden.items():
            if str(insiden[id_insiden]["id_counter"]) == hapus_id:
                del insiden[id_insiden]
                print("Insiden berhasil dihapus.")
                ketemu = True
                break
        if not ketemu:

```

```

        print("ID tidak ketemu.")
        input("Tekan Enter untuk melanjutkan program")

# Lihat Semua Pengguna (Admin)
elif role == "admin" and pilih_menu == "7":
    os.system('cls')
    print("===== Lihat Semua Pengguna =====")
    if len(pengguna) == 0:
        print("Belum ada pengguna terdaftar.")
    else:
        for username, data in pengguna.items():
            print(f"Username: {username}, Role: {data['role']}")

        input("Tekan Enter untuk melanjutkan program")

# Statistik Insiden (Admin)
elif role == "admin" and pilih_menu == "8":
    os.system('cls')
    print("===== Statistik Insiden =====")
    if len(insiden) == 0:
        print("Belum ada insiden tercatat.")
    else:
        jumlah_status = {}
        for id_insiden, data in insiden.items():
            status = data["status"]
            if status in jumlah_status:
                jumlah_status[status] += 1
            else:
                jumlah_status[status] = 1
        for status, jumlah in jumlah_status.items():
            print(f"{status}: {jumlah} insiden")

        input("Tekan Enter untuk lanjut")

else:
    print("Pilihan tidak valid.")
    input("Tekan Enter untuk melanjutkan")

```

#### 4. Hasil Output

```
=== SISTEM MANAJEMEN INSIDEN (SIEM) ===  
1. Login  
2. Register  
3. Keluar  
Pilih menu: 2  
=== REGISTER ===  
Masukkan username anda: lita  
Masukkan password: 189  
Selamat registrasi berhasil  
Tekan Enter untuk lanjut ke menu
```

Gambar 4.1 Register berhasil

```
===== REGISTER =====  
Masukkan username anda: diftya  
Masukkan password: 016  
Username ini tidak bisa dipakai.  
Tekan Enter untuk lanjut ke menu
```

Gambar 4.2 Register gagal

```
=== MENU UTAMA (ADMIN) ===  
Login sebagai: diftya (admin)  
1. Tambah Insiden  
2. Lihat Insiden  
3. Ubah Status Insiden  
4. Logout  
5. Filter Insiden berdasarkan Status  
6. Hapus Insiden  
7. Lihat Semua Pengguna  
8. Statistik Insiden  
Pilih menu: 4  
Logout berhasil.  
Tekan Enter untuk kembali ke menu awal
```

Gambar 4.3 Login berhasil dan muncul menu (admin)

```

===== LOGIN =====
Username: animehj
Password: 042
Username atau passwordmu salah
Tekan Enter untuk melanjutkan program

```

Gambar 4.4 Login gagal

```

===== Tambah Insiden =====
Judul insiden: kebocoran aplikasi A
Tingkat keparahan: high
Tanggal (DD/MM/YYYY): 28 Oktober 2024
Insiden berhasil ditambahkan!
Tekan Enter untuk melanjutkan

```

Gambar 4.5 Fitur tambah insiden

```

===== Lihat Daftar Insiden =====
ID: 1, Judul: , Tingkat: , Status: closed, Tanggal: , Pemilik: diftya
Tekan Enter untuk melanjutkan program

```

Gambar 4.6 Fitur lihat insiden

```

===== Mengubah Status Insiden =====
Masukkan ID insiden: 1
Status sekarang: open
Pilihan status: {'open': 'Insiden baru dilaporkan', 'in progress': 'Insiden sedang ditangani', 'resolved': 'Insiden berhasil diselesaikan', 'closed': 'Kasus telah ditutup'}
Masukkan status baru: closed
Status berhasil diubah!
Tekan Enter untuk melanjutkan

```

Gambar 4.7 Fitur ubah status insiden

```

===== MENU UTAMA (ADMIN) =====
Login sebagai: diftya (admin)
1. Tambah Insiden
2. Lihat Insiden
3. Ubah Status Insiden
4. Logout
5. Filter Insiden berdasarkan Status
6. Hapus Insiden
7. Lihat Semua Pengguna
8. Statistik Insiden
Pilih menu: 4
Logout berhasil.
Tekan Enter untuk kembali ke menu awal

```

Gambar 4.8 Fitur Logout

```
===== Filter Status Insiden =====  
Masukkan status untuk filter: open  
ID: 2, Judul: jika kau melihat aku begini, Status: open, Pemilik: diftya  
Tekan Enter untuk lanjut
```

Gambar 4.9 Fitur filter status insiden

```
===== Hapus Insiden =====  
Masukkan ID insiden yang ingin dihapus: 1  
Insiden berhasil dihapus.  
Tekan Enter untuk melanjutkan program
```

Gambar 4.10 Fitur hapus insiden

```
===== Lihat Semua Pengguna =====  
Username: serigala jika, Role: user  
Tekan Enter untuk melanjutkan program
```

Gambar 4.11 Fitur lihat semua pengguna

```
===== Statistik Insiden =====  
open: 2 insiden  
Tekan Enter untuk lanjut
```

Gambar 4.12 Fitur lihat statistik insiden

## 5. Langkah-langkah GIT

### 5.1 GIT Add

```
PS C:\Users\ASUS\Documents\Kuliah\Praktikum-APD\post-test\post-test-apd-5> git add .
```

## 5.2 GIT Commit

```
PS C:\Users\ASUS\Documents\Kuliah\Praktikum-APD\post-test\post-test-apd-5> git add .
PS C:\Users\ASUS\Documents\Kuliah\Praktikum-APD\post-test\post-test-apd-5> git commit -m "revisi dan penambahan"
[main 7b853b6] revisi dan penambahan
1 file changed, 49 insertions(+), 83 deletions(-)
```


## 5.3 GIT Remote



```
PS C:\Users\ASUS\Documents\Kuliah\Praktikum-APD\post-test\post-test-apd-5> git remote -v
origin https://github.com/HikaruYui/Praktikum-APD.git (fetch)
origin https://github.com/HikaruYui/Praktikum-APD.git (push)
```



## 5.4 GIT Push



```
PS C:\Users\ASUS\Documents\Kuliah\Praktikum-APD\post-test\post-test-apd-5> git push origin main
To https://github.com/HikaruYui/Praktikum-APD.git
```

## 5.4 Github

Praktikum-APD / post-test / post-test-apd-6 / 

 HikaruYui Delete post-test/post-test-apd-6/README.md b519573 · 1 minute ago  History

Name	Last commit message	Last commit date
 ..		
 2509106042-Diftya_Azzahra-PT-6.py	revisi dan penambahan	1 minute ago