

**FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO**  
**UNIVERSITAS TEKNOLOGI YOGYAKARTA**  
**UJIAN AKHIR SEMESTER GANJIL TAHUN AKADEMIK 2021/2022**

---

PROGRAM STUDI : S1 Informatika Kelas G

Mata Ujian	: Algoritma Pemrograman
Hari/Tanggal	: Senin, 20 Januari 2022
Dosen	: Canggi Puspo Wibowo, S.T., M.Eng.
Sifat	: <b><i>Take home, buka buku dan referensi di internet</i></b>

---

**Anggota Kelompok :**

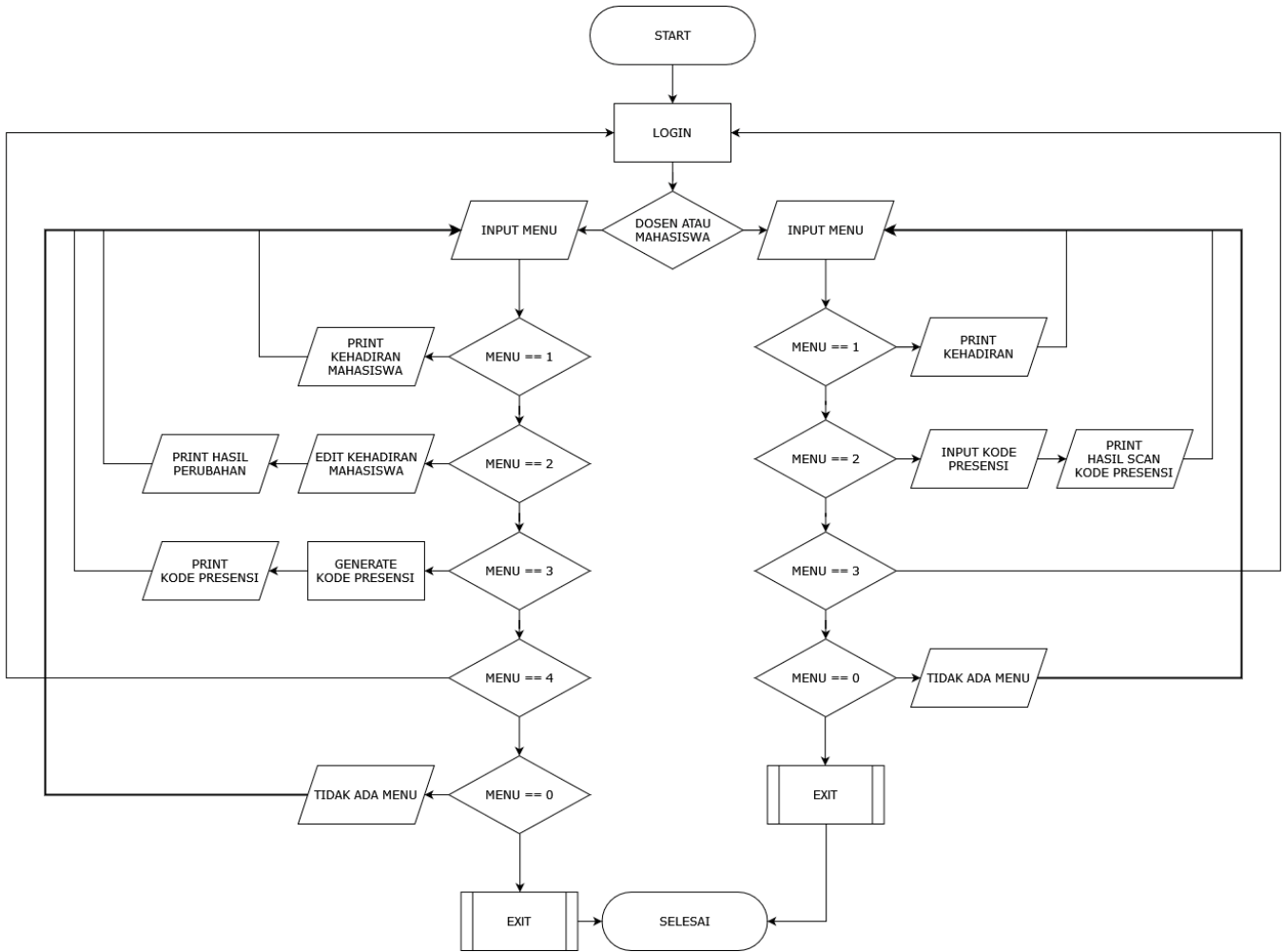
1. 5210411383 | Allan Bil Faqih
2. 5210411332 | Nur Ilham Syah
3. 5210411384 (Keluar dari UTY)

**Laporan :**

A. Algoritma

1. Start
2. User bisa melakukan *Login*.
3. Apakah user *Dosen* atau *Mahasiswa*?
  - a. Jika Dosen maka user bisa memilih menu **1** sampai **4** dan juga **0**.
    - i. Menu **1** Dosen bisa menampilkan Kehadiran mahasiswa, lalu kembali ke pilihan menu (kembali ke *step-a*).
    - ii. Menu **2** Dosen bisa merubah kehadiran mahasiswa dan menampilkan hasil editannya, lalu kembali ke pilihan menu (kembali ke *step-a*).
    - iii. Menu **3** Dosen bisa membuat/generate kode presensi untuk mahasiswa dan menampilkan hasil generate kode tersebut, lalu kembali ke pilihan menu (kembali ke *step-a*).
    - iv. Menu **4** Dosen bisa kembali ke menu Login (kembali ke *step-2*).
    - v. Menu **0** Dosen bisa keluar dari aplikasi (kembali ke *step-4*).
    - vi. Lainnya Dosen dikembalikan ke menu pilihan (kembali ke *step-a*).
  - b. Jika Mahasiswa maka user bisa memilih menu **1** sampai **3** dan juga **0**.
    - i. Kurangi Menu **1** Mahasiswa bisa menampilkan Kehadiran, lalu kembali ke pilihan menu (kembali ke *step-b*).
    - ii. Menu **2** Mahasiswa bisa scan kode presensi, lalu kembali ke pilihan menu (kembali ke *step-b*).
    - iii. Menu **3** Mahasiswa bisa kembali ke menu Login (kembali ke *step-2*).
    - iv. Menu **0** Mahasiswa bisa keluar dari aplikasi (kembali ke *step-4*).
    - v. Lainnya Mahasiswa dikembalikan ke menu pilihan (kembali ke *step-b*).
4. Selesai

## B. Flowchart



### C. Source Code

#### 1. Fungsi squek()

##### Codes

```
10 def squek():
11 > def createTable(): ...
105
106 > def dataCount(): ...
133
134 > def exampleData(): ...
215
216 createTable()
```

##### Penjelasan

Fungsi ini digunakan sebagai *parent* dari fungsi fungsi lainnya.

#### 2. Fungsi createTable()

##### Codes

```
def createTable():
    cur.execute("""CREATE TABLE IF NOT EXISTS "student" (
        "ID" INTEGER,
        "name" TEXT NOT NULL,
        "phone" TEXT NOT NULL,
        "alamat" TEXT NOT NULL,
        "userID" INTEGER NOT NULL,
        PRIMARY KEY("ID" AUTOINCREMENT),
        FOREIGN KEY("userID") REFERENCES [account] (ID)
    );""")

    cur.execute("""CREATE TABLE IF NOT EXISTS "teacher" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "account" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "course" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "academic_year" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "semester" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "attendance_header" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "attendance_detail" ( ...
    cur.execute("""CREATE TABLE IF NOT EXISTS "meeting_code" ( ...
    dataCount()
```

##### Penjelasan

Fungsi ini digunakan untuk membuat table pada database, seperti student, teacher, account, course, academic\_year, semester, attendance\_header, attendance\_detail, dan meeting\_code.

### 3. Fungsi dataCount()

#### Codes

```
def dataCount():
    cur.execute("select count(*) as count from student")
    for r1 in cur.fetchall():
        count1 = r1[0]
    cur.execute("select count(*) as count from teacher")
    for r2 in cur.fetchall(): ...
    cur.execute("select count(*) as count from account")
    for r3 in cur.fetchall(): ...
    cur.execute("select count(*) as count from course")
    for r4 in cur.fetchall(): ...
    cur.execute("select count(*) as count from semester")
    for r5 in cur.fetchall(): ...
    cur.execute("select count(*) as count from attendance_header")
    for r6 in cur.fetchall(): ...
    cur.execute("select count(*) as count from attendance_detail")
    for r7 in cur.fetchall(): ...

    countAll = [count1, count2, count3, count4, count5, count6, count7]

    if(all(v == 0 for v in countAll)):
        exampleData()
```

#### Penjelasan

Fungsi ini digunakan untuk mengecek apakah ada data pada table student, teacher, account, course, semester, attendance\_header, dan attendance\_detail, lalu jika kosong akan diisi dengan data example/contoh yang saya buat.

### 4. Fungsi exampleData()

#### Codes

```
def exampleData():
    cur.execute("""INSERT INTO "student"
VALUES (1,'Allan Bil Faqih','698521421442','7th Heaven, Wutai',2),
(2,'Zack Fair','4205557892102','Sector 5, Midgar',4),
(3,'Linus Sebastian','666251778921','Vancouver, Canada',6),
(4,'Thio Joe','178942314979','Detroit, USA',8);
""")
    cur.execute("""INSERT INTO "teacher" ...
    cur.execute("""INSERT INTO "account" ...
    cur.execute("""INSERT INTO "course" ...
    cur.execute("""INSERT INTO "academic_year" ...
    cur.execute("""INSERT INTO "semester" (name) ...
    cur.execute("""INSERT INTO "attendance_header" ...
    cur.execute("""INSERT INTO "attendance_detail" ...

    conn.commit()
```

#### Penjelasan

Fungsi ini digunakan untuk menambahkan data example ke dalam table student, teacher, account, course, semester, attendance\_header, dan attendance\_detail.

### 5. Fungsi main()

#### Codes

```

def main():
    main.userID = 0
    print("\n[ SISTEM PRESENSI ]")

    def auth(username, password): ...

    def login(): ...

    def pageTeacher(): ...

    def pageStudent(): ...

    def showAtdforStd(): ...

    def processingCode(startDate, endDate, numMeet, courseID): ...

    def scanCode(): ...

    def showAtdforTch(): ...

    def updateingAtd(numMeet, textMeet, studName): ...

    def editAtd(): ...

    def createCode(): ...

```

### Penjelasan

Fungsi ini digunakan sebagai parent dari fungsi fungsi lainnya.

## 6. Fungsi auth()

### Codes

```
def main():
    main.userID = 0
    print("\n[ SISTEM PRESENSI ]")

    def auth(username, password):
        cur.execute("""select count(*) from account
                        where username=? and password=?""", (username, password),)
        for row in cur.fetchall():
            usercount = row[0]

        if(usercount == 0):
            print("[!] Username atau Password salah!")
            return login()
        else:
            cur.execute("select * from account where username=?", (username,))
            for row in cur.fetchall():
                status = row[4]

            if(status == 1):
                cur.execute("""select teacher.ID, teacher.name, account.ID,
                                account.username, account.password, account.type
                                from teacher
                                left join account on teacher.userID = account.ID
                                where account.username=?""", (username,))
                for row in cur.fetchall():
                    main.userID = row[0]
                print("\n[i] Selamat Datang Dosen.")
                pageTeacher()

            elif(status == 0):
                cur.execute("""select student.ID, student.name, account.ID,
                                account.username, account.password, account.type
                                from student
                                left join account on student.userID = account.ID
                                where account.username=?""", (username,))
                for row in cur.fetchall():
                    main.userID = row[0]
                print("\n[i] Selamat Datang Mahasiswa")
                pageStudent()
```

### Penjelasan

Fungsi ini digunakan untuk melakukan login, setelah user memasukkan username dan password, program ini akan mengambil data dari database lalu akan mengecek apakah si user ini dosen atau mahasiswa lalu akan memanggil fungsi sesuai dengan peran user.

## 7. Fungsi login()

### Codes

```
def login():
    print("\n[i] Login Page ")
    inpUsername = input("[=] Input Username : ")
    inpPassword = input("[=] Input Password : ")

    if(inpUsername == "" or inpPassword == ""):
        print("[!] Input Tidak boleh kosong")
        return login()
    else:
        auth(inpUsername, inpPassword)
```

### Penjelasan

Fungsi ini digunakan untuk melakukan login, pertama user perlu memasukkan username dan password, lalu akan memanggil fungsi auth().

## 8. Fungsi pageTeacher()

### Codes

```
def pageTeacher():
    studMenu = ["[1] LIHAT KEHADIRAN MAHASISWA",
                "[2] EDIT KEHADIRAN MAHASISWA",
                "[3] BUAT KODE PRESENSI",
                "[4] KEMBALI KE LOGIN",
                "[0] KELUAR APLIKASI"]

    print("\n[i] Pilih Menu. \n")

    for i in studMenu:
        print(i)

    selectMenu = input("\n[=] Masukkan Angka : ")

    if(selectMenu == "1"):
        showAtdforTch()
        return pageTeacher()
    elif(selectMenu == "2"):
        editAtd()
        return pageTeacher()
    elif(selectMenu == "3"):
        createCode()
        return pageTeacher()
    elif(selectMenu == "4"):
        return login()
    elif(selectMenu == "0"):
        print("[i] Goodbye!")
        return
    else:
        print("[!] Input Salah!")
```

### Penjelasan

Fungsi ini digunakan untuk mengeluarkan menu jika user merupakan dosen, lalu user dapat memilih opsi di dalam menu tersebut.

## 9. Fungsi pageStudent()

### Codes

```
def pageStudent():
    studMenu = ["[1] LIHAT KEHADIRAN",
                "[2] SCAN KODE PRESENSI",
                "[3] KEMBALI KE LOGIN",
                "[0] KELUAR APLIKASI"]

    print("\n[i] Pilih Menu. \n")

    for i in studMenu:
        print(i)

    selectMenu = input("\n[=] Masukkan Angka : ")

    if(selectMenu == "1"):
        showAtdforStd()
        return pageStudent()
    elif(selectMenu == "2"):
        scanCode()
        return pageStudent()
    elif(selectMenu == "3"):
        return login()
    elif(selectMenu == "0"):
        print("[i] Goodbye!")
        return
    else:
        print("[!] Input Salah!")
```

### Penjelasan

Fungsi ini digunakan untuk mengeluarkan menu jika user merupakan mahasiswa, lalu user dapat memilih opsi di dalam menu tersebut.

### 10. Fungsi showAtdforStd()

#### Codes

```
def showAtdforStd():
    thisYear = dt.datetime.now()
    lastYear = dt.datetime(thisYear.year - 1, thisYear.month, thisYear.day,
                           thisYear.hour, thisYear.minute, thisYear.second, thisYear.microsecond)

    cur.execute("select * from attendance_header where studentID=?", (main.userID,))
    for row in cur.fetchall():
        getSemes = row[1]
        getHeaderID = row[0]

    print("\n[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik ",
          lastYear.year, "/", thisYear.year, "- Semester", getSemes)

    colAtd = ['Mata Kuliah', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14']
    cur.execute("""select course.name, meet1, meet2,
                    meet3, meet4, meet5, meet6, meet7,
                    meet8, meet9, meet10, meet11,
                    meet12, meet13, meet14 from attendance_detail
                    left join course on attendance_detail.courseID = course.ID
                    where headerID=?""", (getHeaderID,))

    newPT = pt()
    newPT.field_names = colAtd
    for r in cur.fetchall():
        newPT.add_row(r)
    print(newPT)
```

### Penjelasan

Fungsi ini digunakan untuk menampilkan kehadiran mahasiswa untuk user(mahasiswa), pertama dia akan mengambil data dari table attendance\_header seperti semester dan headerID, lalu fungsi ini akan menampilkan tahun akademik, semester dan kehadiran mahasiswa dalam matkul yang dia ambil.

### 11. Fungsi processingCode()

#### Codes



```
def processingCode(startDate, endDate, numMeet, courseID):
    format = "%Y-%m-%d %H:%M:%S.%f"
    if(dt.datetime.now() <= dt.datetime.strptime(startDate, format)):
        print("[!] Tidak Waktunya untuk Presensi.")

    elif(dt.datetime.now() >= dt.datetime.strptime(startDate, format) and dt.datetime.now() <= dt.datetime.strptime(endDate, format)):
        getMeet = "meet" + str(numMeet)
        cur.execute("""select * from attendance_header
                    where studentID=?""", (main.userID,))
        for i in cur.fetchall():
            getHeaderID = i[0]

        try:
            cur.execute("""update attendance_detail set {whatMeet}=?
                        where courseID=? and headerID=?""".format(whatMeet=getMeet), ("HADIR", courseID, getHeaderID))
        except:
            print("[!] Ada Kesalahan!")
            return scanCode()
        finally:
            conn.commit()

    elif(dt.datetime.now() >= dt.datetime.strptime(endDate, format)):
        print("[!] Waktu Presensi Habis")
```

### Penjelasan

Fungsi ini digunakan untuk memproses kode yg dimasukkan user(mahasiswa) di dalam fungsi scanCode() di dalam fungsi processingCode() ini si program akan mengecek tanggal dari kode presensi apakah sudah hangus, atau apakah belum dalam waktu presensi, apabila waktunya sesuai maka fungsi ini mengupdate data yang ada kehadiran mahasiswa.

## 12. Fungsi scanCode()

### Codes

```
def scanCode():
    print("[i] Scan Presensi | Ketik 'Exit' atau '0' untuk kembali")
    inpCode = input("[=] Silakan Input Kode yang sudah di share oleh Dosen Pengampu : ")

    if(inpCode == "exit" or inpCode == "Exit" or inpCode == "0"):
        return pageStudent()

    try:
        cur.execute("select * from meeting_code where storedCode=?", (inpCode,))
    except:
        print("[!] Ada Kesalahan!")
        return scanCode()
    finally:
        for r in cur.fetchall():
            getCourseID = r[1]
            #getCode = r[2]
            getNumMeeting = r[3]
            getStartDate = r[4]
            getEndDate = r[5]
            print(getStartDate, getEndDate, dt.datetime.now())
            processingCode(getStartDate, getEndDate, getNumMeeting, getCourseID)
```

### Penjelasan

Fungsi ini digunakan untuk melakukan scan kode presensi, mahasiswa harus mengisi input dengan kode presensi, lalu fungsi ini akan memanggil data dari table meeting\_code, data yang diambil merupakan ID dari matkul, meeting ke berapa, waktu mulai dapat presensi, waktu dan hangusnya kode presensi, setelah mendapatkan data maka akan memanggil fungsi processingCode().

## 13. Fungsi showAtdforTch()

### Codes

```

def showAtdforTch():
    thisYear = dt.datetime.now()
    lastYear = dt.datetime(thisYear.year - 1, thisYear.month, thisYear.day,
                           thisYear.hour, thisYear.minute, thisYear.second, thisYear.microsecond)

    cur.execute("""select * from attendance_detail
                  left join course on attendance_detail.courseID = course.ID
                  left join teacher on course.teachID = teacher.ID
                  left join attendance_header on attendance_detail.headerID = attendance_header.ID
                  where course.teachID=?""", (main.userID,))

    for row in cur.fetchall():
        getSemes = row[26]
        getCourseID = row[17]
        getCourseName = row[18]

    print("\n[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik ",
          lastYear.year, "/", thisYear.year, "- Semester", getSemes, "| Mata Kuliah -", getCourseName)

    colAtd = ['Nama Mahasiswa', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14']
    cur.execute("""select student.name,
                        meet1, meet2, meet3, meet4,
                        meet5, meet6, meet7,
                        meet8, meet9, meet10, meet11,
                        meet12, meet13, meet14 from attendance_detail
                  left join attendance_header on attendance_detail.headerID = attendance_header.ID
                  left join student on attendance_header.studentID = student.ID
                  where courseID=?""", (getCourseID,))

    newPT = pt()
    newPT.field_names = colAtd
    for r in cur.fetchall():
        newPT.add_row(r)
    print(newPT)

```

### Penjelasan

Fungsi ini digunakan untuk menampilkan kehadiran mahasiswa untuk user(dosen), pertama dia akan mengambil data dari table attendance\_detail dan attendance\_header dengan menggunakan join clause, lalu fungsi ini akan menampilkan tahun akademik, semester dan kehadiran para mahasiswa dalam matkul yang dosen ampu.

## 14. Fungsi updatingAtd()

### Codes

```

def updateingAtd(numMeet, textMeet, studName):
    cur.execute("""select * from attendance_header
                  left join student on student.ID = attendance_header.studentID
                  where student.name like ?""", (studName,))

    for i in cur.fetchall():
        getHeaderID = i[0]

    getMeet = "meet" + str(numMeet)

    try:
        cur.execute("""update attendance_detail set {whatMeet}=?
                      where headerID=?""".format(whatMeet=getMeet), (textMeet, getHeaderID,))
    except:
        print("[!] Ada Kesalahan!")
        return editAtd()
    finally:
        conn.commit()

```

### Penjelasan

Fungsi ini digunakan untuk merubah data kehadiran yang ada di dalam kehadiran mahasiswa sesuai dengan mata kuliah yang dosen ampu.

## 15. Fungsi editAtd()

### Codes

```

def editAtd():
    showAtdforTch()
    print("[i] Rubah Kehadiran Mahasiswa | Ketik 'Exit' atau '0' di kolom Nama dan kosongkan kolom Meeting untuk kembali")
    inpName = input("\n[=] Masukkan Nama Mahasiswa : ")
    inpMeet = input("[=] Meeting Keberapa? : ")

    if(inpName == "exit" or inpName == "Exit" or inpName == "0"):
        return pageTeacher()

    studMenu = ["\n[1] HADIR", "[2] SAKIT", "[3] IZIN", "[4] ALPHA"]
    for i in studMenu:
        print(i)

    inpChange = input("\n[=] Silahkan Pilih Nomor : ")
    if(inpChange == "1"):
        updateingAtd(inpMeet, "HADIR", inpName)
    elif(inpChange == "2"):
        updateingAtd(inpMeet, "SAKIT", inpName)
    elif(inpChange == "3"):
        updateingAtd(inpMeet, "IZIN", inpName)
    elif(inpChange == "4"):
        updateingAtd(inpMeet, "ALPHA", inpName)
    else:
        print("[!] Input Salah!")
        return editAtd()

```

## Penjelasan

Fungsi ini digunakan sebagai prerequisite untuk merubah data meeting yang ada di dalam kehadiran mahasiswa sesuai dengan mata kuliah yang dosen ampu, pertama user(dosen) perlu memasukkan nama mahasiswa dan meeting keberapa, lalu dosen harus memilih meeting yg terpilih isinya akan diganti menjadi apa, diantara opsi tersebut adalah HADIR, SAKIT, IZIN, dan ALPHA, setelah itu fungsi ini akan memanggil fungsi updatingAtd().

## 16. Fungsi createCode()

### Codes

```
def createCode():
    dateNow = dt.datetime.now()
    dateLater = dateNow + dt.timedelta(minutes = 30)
    cur.execute("select * from course where teachID=?", (main.userID,))

    for r in cur.fetchall():
        getCourseID = r[0]

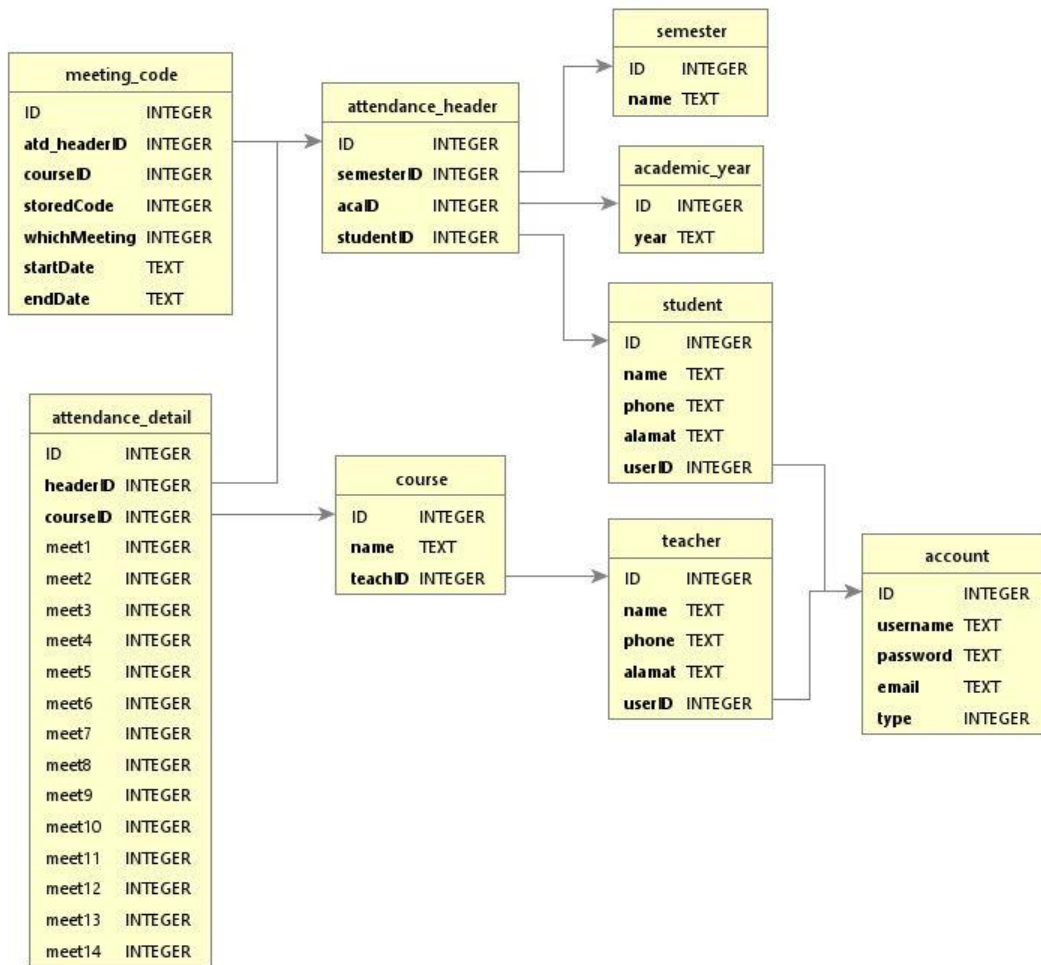
    getCode = ''.join(str(uuid.uuid4()).split('-'))
    print("[i] Buat Kode Untuk Kehadiran Mahasiswa | Ketik 'Exit' atau '0' di kolom Nama untuk kembali")
    print("[i] Kode ini hanya bisa digunakan selama 1 jam dari pembuatan kode ini.")
    inpWhichMeet = int(input("[=] Pilih Meeting ke berapa : "))

    if(inpWhichMeet == "exit" or inpWhichMeet == "Exit" or inpWhichMeet == "0"):
        return pageTeacher()
    try:
        cur.execute("""insert into meeting_code(courseID, storedCode,
                                                whichMeeting, startDate, endDate) values(?, ?, ?, ?, ?)""",
                    (getCourseID, getCode, inpWhichMeet, dateNow, dateLater,))
    except:
        print("[!] Ada Kesalahan")
        return createCode()
    finally:
        print("[i] Kode Yang telah dibuat : ", getCode)
        conn.commit()
```

### Penjelasan

Fungsi ini digunakan untuk membuat kode presensi yang dapat digunakan oleh mahasiswa, pertama program ini akan mengambil ID mata kuliah dari dosen, kemudian program ini akan mengenerate kode secara random, lalu dosen dapat memasukkan kode ini untuk meeting ke berapa, setelah itu kode ini akan disimpan di database dan ditampilkan untuk dikirim ke mahasiswa dosennya.

## D. Entity Relationship Diagram



## E. Hasil Eksekusi

```
[ SISTEM PRESENSI ]

[i] Login Page
[=] Input Username : salvia
[=] Input Password : salvia

[i] Selamat Datang Dosen.

[i] Pilih Menu.

[1] LIHAT KEHADIRAN MAHASISWA
[2] EDIT KEHADIRAN MAHASISWA
[3] BUAT KODE PRESENSI
[4] KEMBALI KE LOGIN
[0] KELUAR APLIKASI

[=] Masukkan Angka : 1

[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik 2021 / 2022 - Semester 1 | Mata Kuliah - Internet Comment Etiquette Kelas Z
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Nama Mahasiswa | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Allan Bil Faqih | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Zack Fair | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Linus Sebastian | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Thio Joe | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

[i] Pilih Menu.

[1] LIHAT KEHADIRAN MAHASISWA
[2] EDIT KEHADIRAN MAHASISWA
[3] BUAT KODE PRESENSI
[4] KEMBALI KE LOGIN
[0] KELUAR APLIKASI

[=] Masukkan Angka : █
```

^tampilan untuk user(dosen) dan opsi LIHAT KEHADIRAN MAHASISWA.

```
[=] Masukkan Angka : 2

[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik 2021 / 2022 - Semester 1 | Mata Kuliah - Internet Comment Etiquette Kelas Z
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Nama Mahasiswa | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Allan Bil Faqih | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Zack Fair | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Linus Sebastian | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Thio Joe | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

[i] Rubah Kehadiran Mahasiswa | Ketik 'Exit' atau '0' di kolom Nama dan kosongkan kolom Meeting untuk kembali

[=] Masukkan Nama Mahasiswa : Thio Joe
[=] Meeting Beberapa? : 11

[1] HADIR
[2] SAKIT
[3] IZIN
[4] ALPHA

[=] Silahkan Pilih Nomor : 4

[i] Pilih Menu.

[1] LIHAT KEHADIRAN MAHASISWA
[2] EDIT KEHADIRAN MAHASISWA
[3] BUAT KODE PRESENSI
[4] KEMBALI KE LOGIN
[0] KELUAR APLIKASI

[=] Masukkan Angka : 1

[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik 2021 / 2022 - Semester 1 | Mata Kuliah - Internet Comment Etiquette Kelas Z
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Nama Mahasiswa | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Allan Bil Faqih | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Zack Fair | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Linus Sebastian | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Thio Joe | None | None | None | None | None | None | None | None | None | None | None | ALPHA | None | None |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

[i] Pilih Menu.
```

^tampilan ketika dosen merubah kehadiran mahasiswa 'Thio Joe' dari None ke ALPHA.

```

[i] Pilih Menu.

[1] LIHAT KEHADIRAN MAHASISWA
[2] EDIT KEHADIRAN MAHASISWA
[3] BUAT KODE PRESENSI
[4] KEMBALI KE LOGIN
[0] KELUAR APLIKASI

[=] Masukkan Angka : 3
[i] Buat Kode Untuk Kehadiran Mahasiswa | Ketik 'Exit' atau '0' di kolom Nama untuk kembali
[i] Kode ini hanya bisa digunakan selama 1 jam dari pembuatan kode ini.
[=] Pilih Meeting ke berapa : 11
[i] Kode Yang telah dibuat : f658e2e2a12f48ef801eff2f8d3b9f12

[i] Pilih Menu.

```

^tampilan ketika dosen mengenerate kode presensi untuk meeting ke 11.

```

[i] Pilih Menu.

[1] LIHAT KEHADIRAN MAHASISWA
[2] EDIT KEHADIRAN MAHASISWA
[3] BUAT KODE PRESENSI
[4] KEMBALI KE LOGIN
[0] KELUAR APLIKASI

[=] Masukkan Angka : 4

[i] Login Page
[=] Input Username : allan
[=] Input Password : nalla

[i] Selamat Datang Mahasiswa

[i] Pilih Menu.

```

^tampilan ketika dosen kembali ke menu login lalu login sebagai mahasiswa.

```

[=] Masukkan Angka : 1

[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik 2021 / 2022 - Semester 1
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Mata Kuliah | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Internet Comment Etiquette Kelas Z | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Stopping Ganon Kelas X | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Sporty Things Kelas Ψ | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
| Try Not Death Kelas ☹️ | None | None | None | None | None | None | None | None | None | None | None | None | None | None |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

[i] Pilih Menu.

```

^tampilan ketika mahasiswa menampilkan kehadirannya pada mata kuliah.

```
[=] Masukkan Angka : 2
[i] Scan Presensi | Ketik 'Exit' atau '0' untuk kembali
[=] Silakan Input Kode yang sudah di share oleh Dosen Pengampu : f658e2e2a12f48ef801eff2f8d3b9f12
2022-01-17 16:59:24.027297 2022-01-17 17:29:24.027297 <built-in method now of type object at 0x00007FFD479DFF10>
```

```
[i] Pilih Menu.
```

```
[1] LIHAT KEHADIRAN
[2] SCAN KODE PRESENSI
[3] KEMBALI KE LOGIN
[0] KELUAR APLIKASI
```

```
[=] Masukkan Angka : 1
```

```
[i] Keaktifan Kehadiran Mahasiswa | Tahun Akademik 2021 / 2022 - Semester 1
```

Mata Kuliah	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Internet Comment Etiquette Kelas Z	None	None	None	None	None	None	None	None	None	None	HADIR	None	None	None
Stopping Ganon Kelas X	None	None	None	None	None	None	None	None	None	None	None	None	None	None
Sporty Things Kelas Ψ	None	None	None	None	None	None	None	None	None	None	None	None	None	None
Try Not Death Kelas %	None	None	None	None	None	None	None	None	None	None	None	None	None	None

^tampilan ketika mahasiswa melakukan scan kode presensi pada matakuliah Internet Comment Etiquette Kelas Z.

```
[i] Pilih Menu.
```

```
[1] LIHAT KEHADIRAN
[2] SCAN KODE PRESENSI
[3] KEMBALI KE LOGIN
[0] KELUAR APLIKASI
```

```
[=] Masukkan Angka : 0
```

```
[i] Goodbye!
```

^tampilan ketika keluar dari aplikasi.