

UAS GUI  
KELOMPOK MUSKETREES

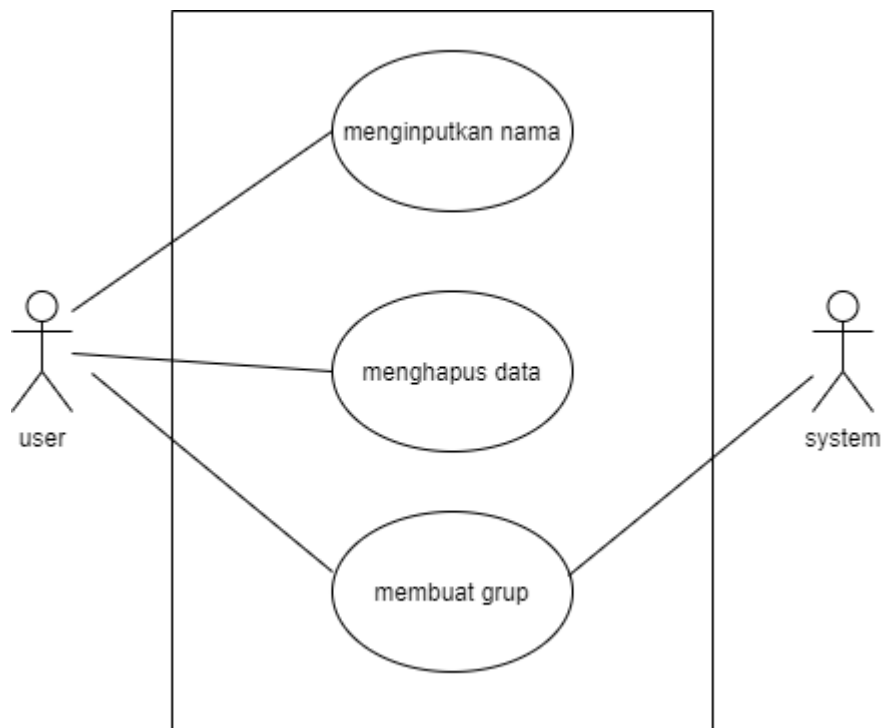
Anggota :

1. Gilang Aditia, 19104043
2. Abdurahman Ibnul Rasidi, 19104032
3. Rahmanda Trinova, 19104051

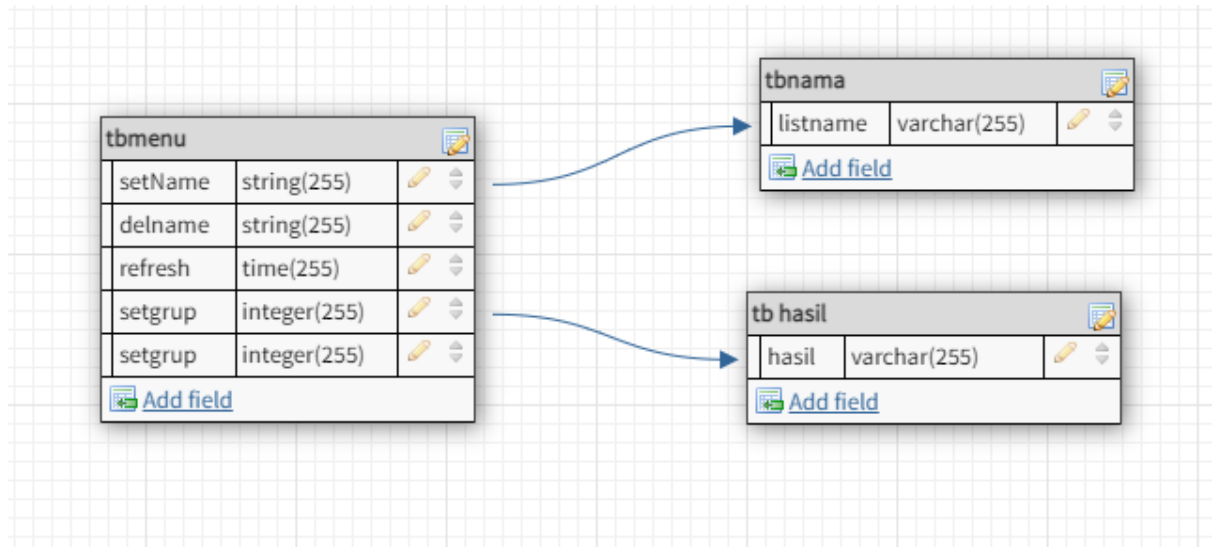
Buatlah sebuah aplikasi dengan menerapkan bahasa pemrograman Python dengan framework PyQT sesuai dengan kasus yang dipilih

Aplikasi Team Random Generator

- a. Use case diagram



b. Desain Database



c. Fitur aplikasi pyqt

Terdapat kolom untuk memasukan nama serta tombol tambah untuk menambahkan nama kedalam list nama dan refres untuk mengatur ulang tampilan list nama apabila ada set nama yang belum masuk lalu ada kolom jumlah grup disitu menginputkan grup yang mau di buat dan hasilnya akan muncul

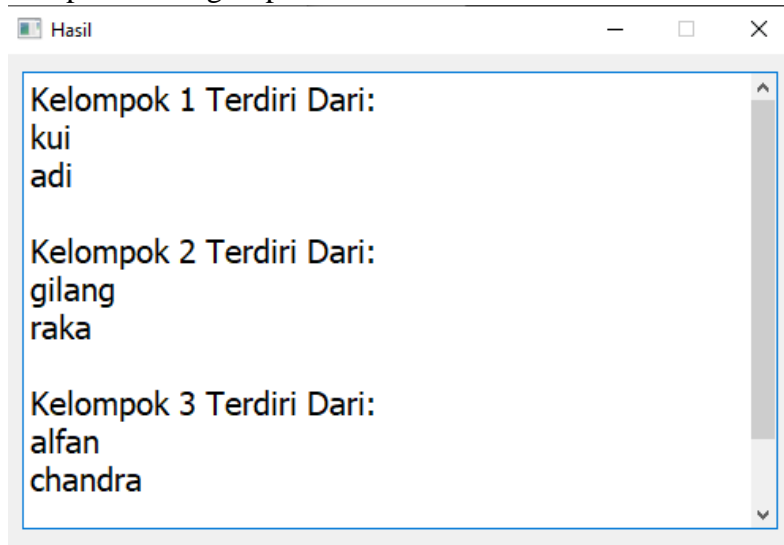
Tampilan awal

Initial application window titled "Form" showing the following interface:

- Input field: masukan nama
- Buttons: tambah, Refresh
- Buttons: Hapus, Hapus Semua
- Input field: jumlah grup
- Button: Buat grup
- Table:

	id	Nama
1	31	gilang
2	32	kui
3	33	alfan
4	34	adi
5	35	chandra
6	36	raka

#### Tampilan hasil group



- d. Tampilan coding yang terhubung dengan database

```
# -*- coding: utf-8 -*-

# Form implementation generated from reading ui file 'untitled.ui'
#
# Created by: PyQt5 UI code generator 5.15.2
#
# WARNING: Any manual changes made to this file will be lost when pyu
ic5 is
# run again. Do not edit this file unless you know what you are doin
g.

from PyQt5 import QtCore, QtGui, QtWidgets
import mysql.connector as mc
import random
from PyQt5.QtWidgets import QApplication, QPlainTextEdit, QWidget, QLi
neEdit, QPushButton, QVBoxLayout, QLabel, QComboBox, QSpinBox

class Hasil(QWidget):
    def __init__(self):
        super().__init__()
        self.setWindowTitle('Hasil')
        self.setFixedWidth(500)
        self.setStyleSheet("""
            QLineEdit{
                font-size: 30px
            }
            QPushButton{
                font-size: 30px
            }
        """)
```

```

        }
        QComboBox{
            font-size: 30px
        }
        QPlainTextEdit{
            font-size: 20px
        }
        """
    mainLayout = QVBoxLayout()

    self.text2 = QPlainTextEdit()
    mainLayout.addWidget(self.text2)

    self.setLayout(mainLayout)

    def closeEvent(self, event):
        self.text2.clear()
        event.accept()

    def displayInfo(self):
        self.show()

class Ui_Form(object):
    def setupUi(self, Form):
        Form.setObjectName("Form")
        Form.resize(583, 467)
        self.secondWin = Hasil()

        self.label = QtWidgets.QLabel(Form)
        self.label.setGeometry(QtCore.QRect(30, 50, 91, 16))
        self.label.setObjectName("label")
        self.lineEdit = QtWidgets.QLineEdit(Form)
        self.lineEdit.setGeometry(QtCore.QRect(30, 70, 231, 20))
        self.lineEdit.setObjectName("lineEdit")

        self.label_3 = QtWidgets.QLabel(Form)
        self.label_3.setGeometry(QtCore.QRect(330, 330, 91, 16))
        self.label_3.setObjectName("label_3")
        self.label_4 = QtWidgets.QLabel(Form)
        self.label_4.setGeometry(QtCore.QRect(440, 40, 91, 16))
        self.label_4.setObjectName("label_4")
        self.pushButton = QtWidgets.QPushButton(Form)
        self.pushButton.setGeometry(QtCore.QRect(30, 100, 75, 23))
        self.pushButton.setObjectName("pushButton")
        self.pushButton.clicked.connect(self.addButtonClick)
        self.pushButton2 = QtWidgets.QPushButton(Form)
        self.pushButton2.setGeometry(QtCore.QRect(130, 100, 75, 23))
        self.pushButton2.setObjectName("pushButton2")

```

```

self.pushButton2.clicked.connect(self.selectData)

self.pushButton_2 = QtWidgets.QPushButton(Form)
self.pushButton_2.setGeometry(QtCore.QRect(30, 170, 75, 20))
self.pushButton_2.setObjectName("pushButton_2")
self.pushButton_2.clicked.connect(self.hapus)
self.pushButton_2.clicked.connect(self.selectData)

self.pushButton_5 = QtWidgets.QPushButton(Form)
self.pushButton_5.setGeometry(QtCore.QRect(130, 170, 75, 20))
self.pushButton_5.setObjectName("pushButton_5")
self.pushButton_5.clicked.connect(self.hapusAll)
self.pushButton_5.clicked.connect(self.selectData)

self.label_4 = QtWidgets.QLabel(Form)
self.label_4.setGeometry(QtCore.QRect(30, 250, 91, 16))
self.label_4.setObjectName("label_4")
self.lineEdit_3 = QtWidgets.QLineEdit(Form)
self.lineEdit_3.setGeometry(QtCore.QRect(30, 270, 231, 20))
self.lineEdit_3.setObjectName("lineEdit_3")

self.pushButton_4 = QtWidgets.QPushButton(Form)
self.pushButton_4.setGeometry(QtCore.QRect(170, 360, 75, 23))
self.pushButton_4.setObjectName("pushButton_4")
self.pushButton_4.clicked.connect(self.randomCek)
self.tableWidget = QtWidgets.QTableWidget(Form)
self.tableWidget.setGeometry(QtCore.QRect(325, 71, 231, 251))
self.tableWidget.setRowCount(3)
self.tableWidget.setColumnCount(2)
self.tableWidget.setObjectName("tableWidget")
self.tableWidget.setColumnWidth(0,15)
self.tableWidget.setHorizontalHeaderLabels(['id','Nama'])

self.retranslateUi(Form)
QtCore.QMetaObject.connectSlotsByName(Form)

mydb = mc.connect(
    host="localhost",
    user="root",
    password="",
    database="randomgg"

)
mycursor = mydb.cursor()
mycursor.execute("SELECT id,nama from hasil")
result = mycursor.fetchall()
self.tableWidget.setRowCount(0)

```

```

        for row_number,row_data in enumerate(result):
            self.tableWidget.insertRow(row_number)
            for column_number,data in enumerate(row_data):
                self.tableWidget.setItem(row_number,column_number,QTableWidgetItem(str(data)))

    def retranslateUi(self, Form):
        _translate = QtCore.QCoreApplication.translate
        Form.setWindowTitle(_translate("Form", "Form"))
        self.label.setText(_translate("Form", "masukan nama"))

        self.label_4.setText(_translate("Form", "Jumlah group : "))
        self.pushButton.setText(_translate("Form", "tambah"))
        self.pushButton2.setText(_translate("Form", "Refresh"))
        self.pushButton_2.setText(_translate("Form", "Hapus"))
        self.pushButton_5.setText(_translate("Form", "Hapus Semua"))
        self.label_4.setText(_translate("Form", "jumlah grup"))
        self.pushButton_4.setText(_translate("Form", "Buat grup"))

    def addButtonClick(self):
        try:
            nama = self.lineEdit.text()
            mydb = mc.connect(
                host="localhost",
                user="root",
                password="",
                database="randomgg"
            )

            mycursor = mydb.cursor()

            query = "INSERT INTO hasil (nama) VALUES (%s)"
            value = (nama,)
            mycursor.execute(query,value)
            mydb.commit()
            self.lineEdit.clear()
            self.selectData()
        except mc.Error as e:
            print('gagal')
            print(nama)

    def selectData(self):
        mydb = mc.connect(
            host="localhost",
            user="root",
            password="",
            database="randomgg"

```

```

    )
    mycursor = mydb.cursor()
    mycursor.execute("SELECT id,nama from hasil")
    result = mycursor.fetchall()
    self.tableWidget.setRowCount(0)

    for row_number,row_data in enumerate(result):
        self.tableWidget.insertRow(row_number)
        for column_number,data in enumerate(row_data):
            self.tableWidget.setItem(row_number,column_number,QtW
idgets.QTableWidgetItem(str(data)))

    def hapus(self):

        try:
            row = self.tableWidget.currentRow()
            thing = self.tableWidget.item(row,0)
            thing1 = self.tableWidget.item(row,1)
            if thing is not None or thing1 is not None and thing.text
() != '':
                currentid = (self.tableWidget.item(row, 0).text() )
                mydb = mc.connect(
                    host="localhost",
                    user="root",
                    password="",
                    database="randomgg"
                )

                mycursor = mydb.cursor()
                query = "DELETE FROM hasil WHERE id=%s"
                value = (currentid,)
                mycursor.execute(query,value)
                mydb.commit()
        except mc.Error as e:
            print('gagal')

    def hapusAll(self):

        try:
            row = self.tableWidget.currentRow()
            thing = self.tableWidget.item(row,0)
            thing1 = self.tableWidget.item(row,1)

            mydb = mc.connect(
                host="localhost",
                user="root",
                password="",
                database="randomgg"
            )

```

```

    )

    mycursor = mydb.cursor()
    query = "DELETE FROM hasil"
    mycursor.execute(query)
    mydb.commit()
except mc.Error as e:
    print('gagal')

def randomCek(self):
    try:

        mydb = mc.connect(
            host="localhost",
            user="root",
            password="",
            database="randomgg"

        )
        mycursor = mydb.cursor()
        mycursor.execute("SELECT nama from hasil")
        result = mycursor.fetchall()
        item = []
        participants = [item[0] for item in result]
        members = int(self.lineEdit_3.text())
        random.shuffle(participants)
        for i in range(len(participants) // members + 1):
            self.secondWin.text2.appendPlainText('')
            print('=====')
            print('Kelompok {} Terdiri Dari:'.format(i + 1))
            group = participants[i*members:i*members + members]
            self.secondWin.text2.appendPlainText('Kelompok {} Terdiri Dari:'.format(i + 1))
            for participant in group:
                print(participant)
                self.secondWin.text2.appendPlainText(participant)
                self.secondWin.displayInfo()
        except mc.Error as e:
            print('gagal')
if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
    Form = QtWidgets.QWidget()
    ui = Ui_Form()
    ui.setupUi(Form)
    Form.show()
    sys.exit(app.exec_())

```