**Aplikasi Sistem Informasi Manajemen Keuangan Administrasi Masjid Berbasis Dekstop**

*Rio Erfian / 18.04.111.00040 Pemrograman Dekstop 3C Teknik Informatika – Universitas Trunojoyo Madura*

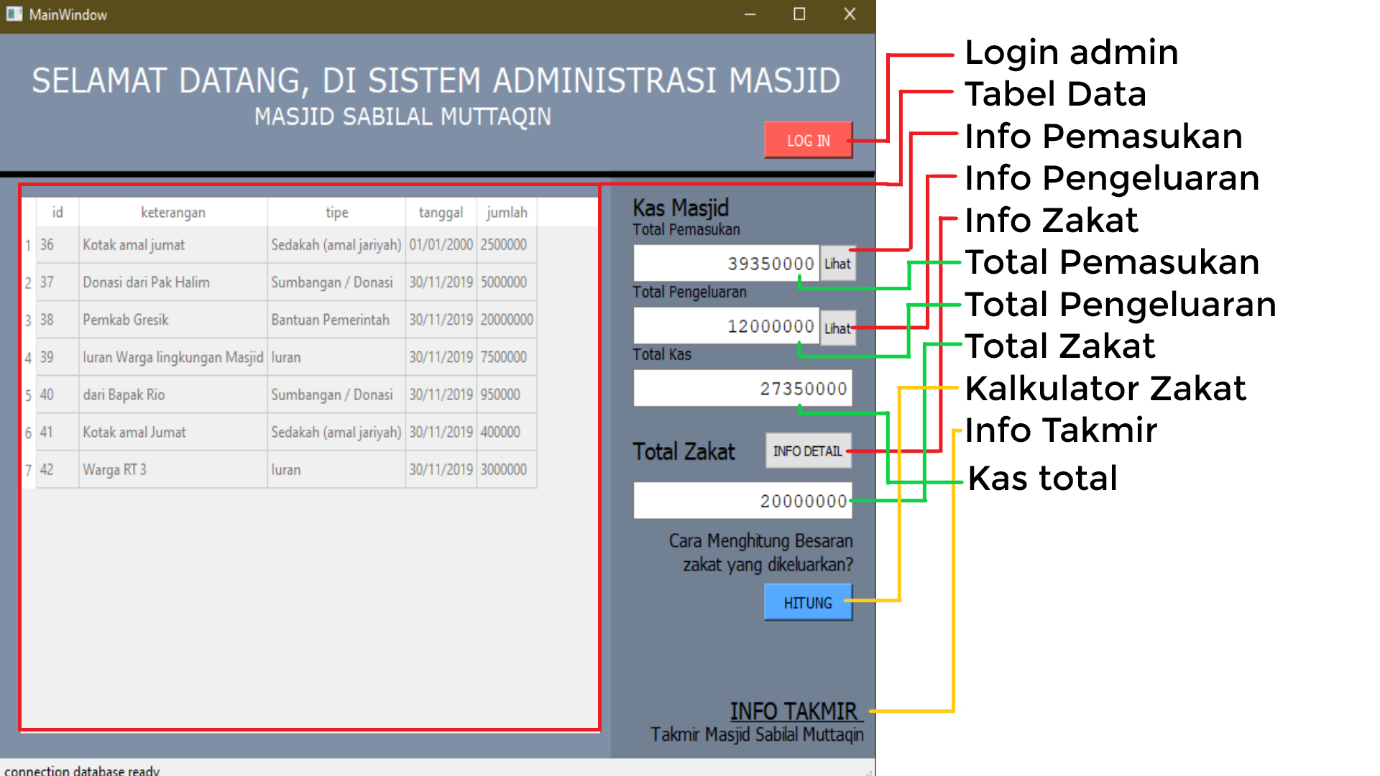
1. **Latar Belakang**

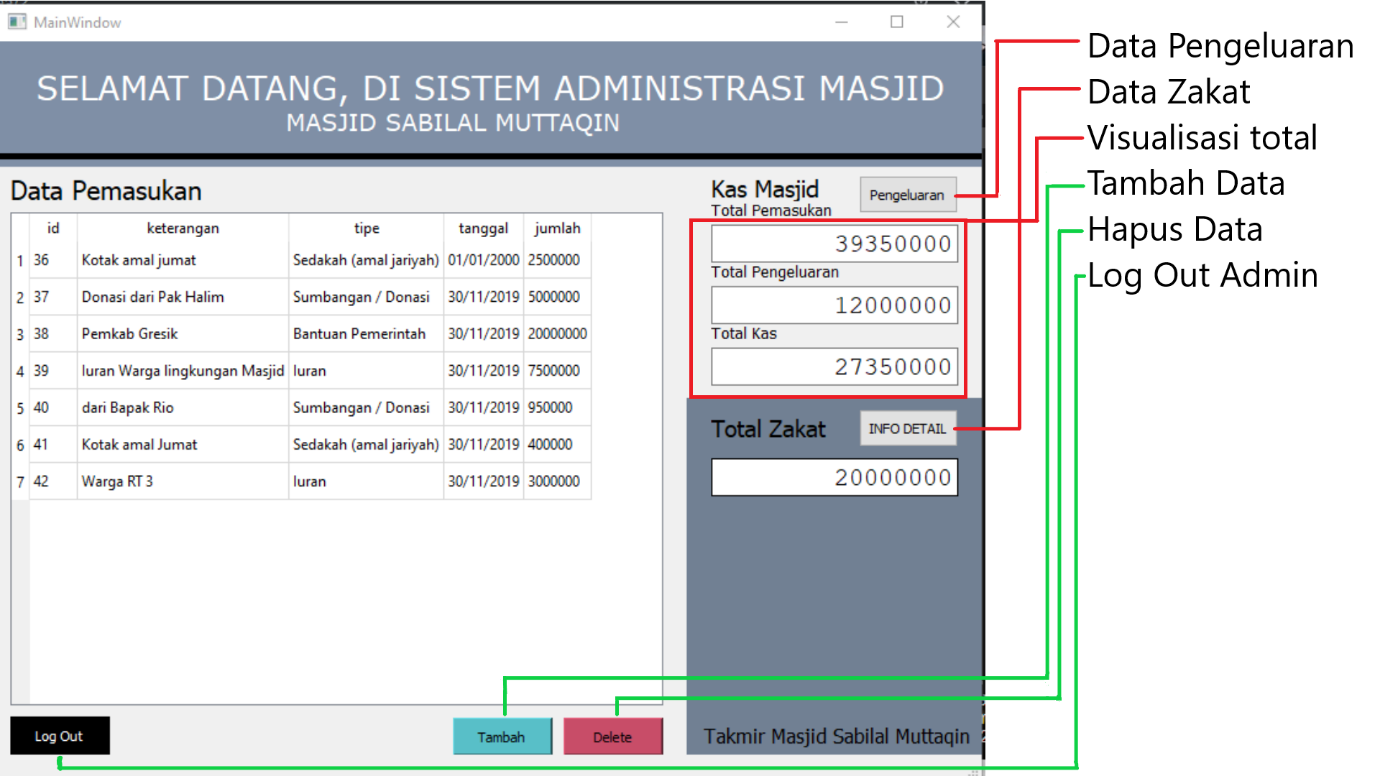
Masjid adalah merupakan rumah ibadah bagi umat islam, dan Masjid juga merupakan entitas publik dimana nilai-nilai spiritual Islam dikembangkan, dan pada kenyataannya Masjid yang merupakan entitas public diperlukan suatu pengelolaan yang Transparan, akuntabilitas dan dalam pengelolaannya juga sangat melibatkan publik. Maka untuk itulah Masjid membutuhkan peran sebuah pembukuan dan akuntansi untuk membuat laporan keuangan, hal itu supaya masyarakat dan jamaah tahu tentang proses Keuangan yang terjadi pada pengelolaan tersebut. Selama ini pengurus membuat Laporan keuangan adalah dalam upaya untuk mempertanggung jawabkan serta untuk menjawab kecurigaan yang sering muncul dari para jamaah. Karena sumber dana Masjid biasanya berasal dari masyarakat, dan masyarakat tentu sangat kritis terhadap penggunaan dana-dana yang bersumber dari sumbangan masyarakat maupun sumbangan dari instansi pemerintah dalam rangka pembangunan masjid, maupun kegiatan-kegiatan keagamaan yang diselenggarakan oleh masjid secara rutin. Penggunaan akuntansi dalam rangka menyampaikan laporan keuangan masjid, karena latar belakang kebutuhan masyarakat akan akuntabilitas dan keterbukaan.

Dengan perkembangan teknologi informasi dewasa ini telah membawa pengaruh yang sangat baik bagi perkembangan Ilmu pengetahuan, Dengan perkembangan tersebut teknologi yang merupakan salah satu alat yang dapat membantu manusia untuk mencapai tujuannya. Dengan teknologi informasi ini manusia dapat menangkap informasi terbaru secara cepat dan akurat. Kegiatan pengelolaan data keuangan masjid masih dikelola secara tradisionil dengan sistem pencatatan secara manual.sehingga membutuhkan waktu yang lama, penyimpanan file tersimpan di banyak buku sehingga para pengurus dalam pencarian data-data susah dan lama, terjadinya kehilangan data dikarenakan penyimpanan data bukan dalam satu folder, tentu memperlambat sebuah informasi sebagai pertanggung jawaban. Setelah melihat masalah-masalah tersebut, maka perlu solusi suatu aplikasi agar data kas bisa dikelola dengan baik, cepat, dan menghasilkan bentuk laporan yang lebih baik dengan pengolahan data yang terprogram.

1. **Penjelasan Umum Program**

Program ini dibuat untuk memenuhi kebutuhan pengelolaan informasi tentang masjid khususnya dalam hal keuangan agar terciptanya keuangan yang transparan dan akuntable, karena hal ini melibatkan uang jamaah dimana biasanya sumber dana berasal dari masyarakat. Tentu masyarakat akan kritis terhadap penggunaan dana tersebut. Fiture ini memperlihatkan table pemasukan dan pengeluaran apa saja yang telah dilakukan serta menampilkan jumlah total pemasukan, pengeluaran dan hasil akhir nya. Fiture lainnya yaitu penyaluran zakat mal dan zakat profesi, dimana terdapat kalkulator mana yang wajib zakat atau tidak sesuai dengan ketentuan syariat. Jika masuk kriteria pengguna bisa menyalurkan ke masjid nanti akan dicatat di program tersebut.

1. **Mock up / Tampilan**



1. **Listing Program**
   1. **Code.py** as **mainwindow**

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import adminpengeluaran, adminpemasukan, adminzakat, adding, login, takmir, calczakatmal

## MAIN WINDOW ##

class Ui(QtWidgets.QMainWindow):

    def \_\_init\_\_(self):

        super(Ui, self).\_\_init\_\_()

        self.model = QSqlTableModel()

        uic.loadUi('dasar.ui', self)

        self.show()

        self.openDB()

        self.model = QtSql.QSqlTableModel()

        self.Login.clicked.connect(self.login)

        self.Takmir.clicked.connect(self.showTakmir)

        self.pushButton.clicked.connect(self.infokas)

        self.pushButton\_2.clicked.connect(self.infokaskeluar)

        self.Zakat3\_2.clicked.connect(self.infozakat)

        self.hitung.clicked.connect(self.calc)

        self.displaytable()

        conn=sqlite3.connect('pemasukan\_kas.db')

        c=conn.cursor()

        datamasuk=c.execute("SELECT SUM(jumlah) FROM catatan\_kas  ")

        for x in datamasuk:

            hasil1=x[0]

        self.lineEdit\_2.setText(str(hasil1))

        datakeluar=c.execute("SELECT SUM(jumlah) FROM catatan\_kaskeluar ")

        for y in datakeluar:

            hasil2=y[0]

        self.lineEdit\_3.setText(str(hasil2))

        if hasil2 == None :

            self.lineEdit\_4.setText(str(hasil2))

        if hasil1== None :

            self.lineEdit\_4.setText("-"+str(hasil2))

        else:

            hasil3=hasil1-hasil2

            self.lineEdit\_4.setText(str(hasil3))

        datazakat=c.execute("SELECT SUM(jumlah) FROM catatan\_zakat ")

        for a in datazakat:

            hasil4=a[0]

        self.Zakat2\_2.setText(str(hasil4))

    def openDB(self):

        db = QtSql.QSqlDatabase.addDatabase('QSQLITE')

        db.setDatabaseName('pemasukan\_kas.db')

        if not db.open():

            self.statusBar().showMessage('connection database error')

            return False

        else:

            self.statusBar().showMessage('connection database ready')

            return True

    def displaytable(self):

        self.model.setTable('catatan\_kas')

        self.model.select()

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

    def infokas(self):

        query = QtSql.QSqlQuery("SELECT \* FROM catatan\_kas ")

        self.model.setQuery(query)

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

    def infokaskeluar(self):

        query = QtSql.QSqlQuery("SELECT \* FROM catatan\_kaskeluar ")

        self.model.setQuery(query)

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

    def infozakat(self):

        query = QtSql.QSqlQuery("SELECT \* FROM catatan\_zakat ")

        self.model.setQuery(query)

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

    def showTakmir(self):

        self.info=takmir.UiTakmir()

        self.close()

        self.info.show()

    def login(self):

        self.log=login.UiLogin()

        self.close()

        self.log.show()

    def calc(self):

        self.calcshow=calczakatmal.Uikalkulatorzakatmal()

        self.calcshow.show()

if \_\_name\_\_ == '\_\_main\_\_':

    app = QtWidgets.QApplication(sys.argv)

    window = Ui()

    app.exec\_()

* 1. **Adminpemasukan.py**

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import code, adminpengeluaran,adminzakat,adding

conn=sqlite3.connect('pemasukan\_kas.db')

c=conn.cursor()

class UiAdmin(QtWidgets.QMainWindow):

    def \_\_init\_\_(self):

        super(UiAdmin, self).\_\_init\_\_()

        uic.loadUi('dasaradmin.ui', self)

        self.model = QtSql.QSqlTableModel()

        self.pushButton.clicked.connect(self.showkeluar)

        self.Zakat3\_2.clicked.connect(self.showzakat)

        self.pushButton\_2.clicked.connect(self.add)

        self.pushButton\_3.clicked.connect(self.remove)

        self.pushButton\_5.clicked.connect(self.backtomenu)

        self.displaytable()

        datamasuk=c.execute("SELECT SUM(jumlah) FROM catatan\_kas  ")

        for x in datamasuk:

            hasil1=x[0]

        self.lineEdit\_2.setText(str(hasil1))

        datakeluar=c.execute("SELECT SUM(jumlah) FROM catatan\_kaskeluar ")

        for y in datakeluar:

            hasil2=y[0]

        self.lineEdit\_3.setText(str(hasil2))

        if hasil2 == None :

            self.lineEdit\_4.setText(str(hasil2))

        if hasil1== None :

            self.lineEdit\_4.setText("-"+str(hasil2))

        else:

            hasil3=hasil1-hasil2

            self.lineEdit\_4.setText(str(hasil3))

        datazakat=c.execute("SELECT SUM(jumlah) FROM catatan\_zakat ")

        for a in datazakat:

            hasil4=a[0]

        self.Zakat2\_2.setText(str(hasil4))

    def add(self):

        self.addrow=adding.Uiadding()

        self.addrow.show()

        self.displaytable()

    def showzakat(self):

        self.zakatwindow=adminzakat.UiAdminZakat()

        self.close()

        self.zakatwindow.show()

* 1. **Adminpengeluaran.py**

    def showkeluar(self):

        self.pengeluaranwindow=adminpengeluaran.UiAdminPengeluaran()

        self.close()

        self.pengeluaranwindow.show()

    def displaytable(self):

        self.model.setTable('catatan\_kas')

        self.model.select()

        self.model.setEditStrategy(QtSql.QSqlTableModel.OnFieldChange)

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

        conn=sqlite3.connect('pemasukan\_kas.db')

        c=conn.cursor()

        datamasuk=c.execute("SELECT SUM(jumlah) FROM catatan\_kas  ")

        for x in datamasuk:

            hasil1=x[0]

        self.lineEdit\_2.setText(str(hasil1))

        datakeluar=c.execute("SELECT SUM(jumlah) FROM catatan\_kaskeluar ")

        for y in datakeluar:

            hasil2=y[0]

        self.lineEdit\_3.setText(str(hasil2))

        if hasil2 == None :

            self.lineEdit\_4.setText(str(hasil2))

        if hasil1== None :

            self.lineEdit\_4.setText("-"+str(hasil2))

        else:

            hasil3=hasil1-hasil2

            self.lineEdit\_4.setText(str(hasil3))

    def remove(self):

        self.model.removeRow(self.tableView.currentIndex().row())

        self.displaytable()

    def backtomenu(self):

        self.dasar=code.Ui()

        self.dasar.show()

        self.close()

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import code, adminpemasukan,adminzakat,adding

conn=sqlite3.connect('pemasukan\_kas.db')

c=conn.cursor()

class UiAdminPengeluaran(QtWidgets.QMainWindow):

    def \_\_init\_\_(self):

        super(UiAdminPengeluaran, self).\_\_init\_\_()

        uic.loadUi('dasaradminkeluar.ui', self)

        self.model = QtSql.QSqlTableModel()

        self.pushButton.clicked.connect(self.showmasuk)

        self.Zakat3\_2.clicked.connect(self.showzakat)

        self.pushButton\_2.clicked.connect(self.add)

        self.pushButton\_3.clicked.connect(self.remove)

        self.pushButton\_5.clicked.connect(self.backtomenu)

        self.displaytable()

        datamasuk=c.execute("SELECT SUM(jumlah) FROM catatan\_kas  ")

        for x in datamasuk:

            hasil1=x[0]

        self.lineEdit\_2.setText(str(hasil1))

        datakeluar=c.execute("SELECT SUM(jumlah) FROM catatan\_kaskeluar ")

        for y in datakeluar:

            hasil2=y[0]

        self.lineEdit\_3.setText(str(hasil2))

        if hasil2 == None :

            self.lineEdit\_4.setText(str(hasil2))

        if hasil1== None :

            self.lineEdit\_4.setText("-"+str(hasil2))

        else:

            hasil3=hasil1-hasil2

            self.lineEdit\_4.setText(str(hasil3))

        datazakat=c.execute("SELECT SUM(jumlah) FROM catatan\_zakat ")

        for a in datazakat:

            hasil4=a[0]

        self.Zakat2\_2.setText(str(hasil4))

    def add(self):

        self.addrow=adding.Uiaddingsub()

        self.addrow.show()

    def showzakat(self):

        self.zakatwindow=adminzakat.UiAdminZakat()

        self.close()

        self.zakatwindow.show()

    def showmasuk(self):

        self.pemasukanwindow=adminpemasukan.UiAdmin()

        self.close()

        self.pemasukanwindow.show()

    def displaytable(self):

        self.model.setTable('catatan\_kaskeluar')

        self.model.select()

        self.model.setEditStrategy(QtSql.QSqlTableModel.OnFieldChange)

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

* 1. adminzakat.py

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import code, adminpemasukan, adminpengeluaran, adding

conn=sqlite3.connect('pemasukan\_kas.db')

c=conn.cursor()

class UiAdminZakat(QtWidgets.QMainWindow):

    def \_\_init\_\_(self):

        super(UiAdminZakat, self).\_\_init\_\_()

        uic.loadUi('dasaradminzakat.ui', self)

        self.model = QSqlTableModel()

        self.show()

        self.model = QtSql.QSqlTableModel()

        self.pushButton.clicked.connect(self.showkas)

        self.Zakat3\_2.clicked.connect(self.infozakat)

        self.pushButton\_2.clicked.connect(self.add)

        self.pushButton\_3.clicked.connect(self.remove)

        self.pushButton\_5.clicked.connect(self.backtomenu)

        self.displaytable()

        datamasuk=c.execute("SELECT SUM(jumlah) FROM catatan\_kas  ")

        for x in datamasuk:

            hasil1=x[0]

        self.lineEdit\_2.setText(str(hasil1))

        datakeluar=c.execute("SELECT SUM(jumlah) FROM catatan\_kaskeluar ")

        for y in datakeluar:

            hasil2=y[0]

        self.lineEdit\_3.setText(str(hasil2))

        if hasil2 == None :

            self.lineEdit\_4.setText(str(hasil2))

        if hasil1== None :

            self.lineEdit\_4.setText("-"+str(hasil2))

        else:

            hasil3=hasil1-hasil2

            self.lineEdit\_4.setText(str(hasil3))

    def remove(self):

        self.model.removeRow(self.tableView.currentIndex().row())

        self.displaytable()

    def backtomenu(self):

        self.dasar=code.Ui()

        self.dasar.show()

        self.close()

        datamasuk=c.execute("SELECT SUM(jumlah) FROM catatan\_kas  ")

        for x in datamasuk:

            hasil1=x[0]

        self.lineEdit\_2.setText(str(hasil1))

        datakeluar=c.execute("SELECT SUM(jumlah) FROM catatan\_kaskeluar ")

        for y in datakeluar:

            hasil2=y[0]

        self.lineEdit\_3.setText(str(hasil2))

        if hasil2 == None :

            self.lineEdit\_4.setText(str(hasil2))

        if hasil1== None :

            self.lineEdit\_4.setText("-"+str(hasil2))

        else:

            hasil3=hasil1-hasil2

            self.lineEdit\_4.setText(str(hasil3))

        datazakat=c.execute("SELECT SUM(jumlah) FROM catatan\_zakat ")

        for a in datazakat:

            hasil4=a[0]

        self.Zakat2\_2.setText(str(hasil4))

    def add(self):

        self.addrow=adding.Uiaddingzakat()

        self.addrow.show()

    def infozakat(self):

        query = QtSql.QSqlQuery("SELECT \* FROM catatan\_zakat ")

        self.model.setQuery(query)

        self.tableView.setModel(self.model)

    def showkas(self):

        self.kaswindow=adminpemasukan.UiAdmin()

        self.close()

        self.kaswindow.show()

    def displaytable(self):

        self.model.setTable('catatan\_zakat')

        self.model.select()

        self.model.setEditStrategy(QtSql.QSqlTableModel.OnFieldChange)

        self.tableView.setModel(self.model)

        self.tableView.resizeColumnsToContents()

        datazakat=c.execute("SELECT SUM(jumlah) FROM catatan\_zakat ")

        for a in datazakat:

            hasil4=a[0]

        self.Zakat2\_2.setText(str(hasil4))

    def remove(self):

        self.model.removeRow(self.tableView.currentIndex().row())

        self.displaytable()

    def backtomenu(self):

        self.dasar=code.Ui()

        self.dasar.show()

        self.close()

* 1. **Adding.py**

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import adminpemasukan

from datetime import datetime

class Uiadding(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(Uiadding, self).\_\_init\_\_()

        uic.loadUi('dialogadd.ui', self)

        hari= datetime.now().day

        bulan=datetime.now().month

        tahun=datetime.now().year

        self.setWindowTitle("add")

        self.pushButton.clicked.connect(self.add\_data)

        self.dateEdit.setDateTime(QtCore.QDateTime(QtCore.QDate(tahun, bulan, hari+1), QtCore.QTime(0, 0, 0)))

        self.setWindowModality(Qt.ApplicationModal)

    def add\_data(self):

        ket=self.lineEdit.text()

        jenis=self.comboBox.currentText()

        tanggal=self.dateEdit.text()

        jumlah=self.lineEdit\_2.text()

        query = QtSql.QSqlQuery()

        query.exec\_("INSERT INTO catatan\_kas (keterangan,tipe,tanggal,jumlah) values ('%s','%s','%s','%s')" %(ket,jenis,tanggal,jumlah))

        self.close()

class Uiaddingsub(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(Uiaddingsub, self).\_\_init\_\_()

        uic.loadUi('dialogsub.ui', self)

        hari= datetime.now().day

        bulan=datetime.now().month

        tahun=datetime.now().year

        self.dateEdit.setDateTime(QtCore.QDateTime(QtCore.QDate(tahun, bulan, hari+1), QtCore.QTime(0, 0, 0)))

        self.setWindowTitle("add")

        self.pushButton.clicked.connect(self.add\_data)

        self.setWindowModality(Qt.ApplicationModal)

    def add\_data(self):

        ket=self.lineEdit.text()

        jenis=self.comboBox.currentText()

        tanggal=self.dateEdit.text()

        jumlah=self.lineEdit\_2.text()

        query = QtSql.QSqlQuery()

        query.exec\_("INSERT INTO catatan\_kaskeluar (keterangan,tipe,tanggal,jumlah) values ('%s','%s','%s','%s')" %(ket,jenis,tanggal,jumlah))

        self.close()

* 1. **login.py**

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

import code, adminpemasukan

class UiLogin(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(UiLogin, self).\_\_init\_\_()

        uic.loadUi('login.ui', self)

        self.setWindowTitle("Login")

        self.pushButton\_2.clicked.connect(self.backtomenu)

        self.pushButton.clicked.connect(self.func)

        self.setWindowModality(Qt.ApplicationModal)

    def func(self):

        username="admin"

        password="123"

        inusername=self.lineEdit.text()

        inpassword=self.lineEdit\_2.text()

        if ((inusername == "admin") and (inpassword =="123")):

            self.homeadmin=adminpemasukan.UiAdmin()

            self.homeadmin.show()

            self.close()

        else:

            self.label\_4.setText("error")

    def backtomenu(self):

        self.dasar=code.Ui()

        self.dasar.show()

        self.close()

class Uiaddingzakat(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(Uiaddingzakat, self).\_\_init\_\_()

        uic.loadUi('dialogaddzakat.ui', self)

        hari= datetime.now().day

        bulan=datetime.now().month

        tahun=datetime.now().year

        self.dateEdit.setDateTime(QtCore.QDateTime(QtCore.QDate(tahun, bulan, hari+1), QtCore.QTime(0, 0, 0)))

        self.setWindowTitle("zakat")

        self.pushButton.clicked.connect(self.add\_data)

        self.setWindowModality(Qt.ApplicationModal)

    def add\_data(self):

        ket=self.lineEdit.text()

        jenis=self.comboBox.currentText()

        tanggal=self.dateEdit.text()

        jumlah=self.lineEdit\_2.text()

        query = QtSql.QSqlQuery()

        query.exec\_("INSERT INTO catatan\_zakat (muzakki,tipe,tanggal,jumlah) values ('%s','%s','%s','%s')" %(ket,jenis,tanggal,jumlah))

        self.close()

* 1. **Calczakatmal.py**

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import code

class Uikalkulatorzakatmal(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(Uikalkulatorzakatmal, self).\_\_init\_\_()

        uic.loadUi('kalkulatorzakatmal.ui', self)

        self.setWindowTitle("Kalkulator Zakat Mal")

        self.pushButton.clicked.connect(self.hitung)

        self.pushButton\_2.clicked.connect(self.zakatprofesi)

        self.setWindowModality(Qt.ApplicationModal)

        nsb= 85\*660485

        self.lineEdit\_7.setText(str(nsb))

    def zakatprofesi(self):

        self.showzakatprofesi=Uikalkulatorzakatprofesi()

        self.close()

        self.showzakatprofesi.show()

    def hitung(self):

        input1=self.lineEdit.text()

        input2=self.lineEdit\_2.text()

        input3=self.lineEdit\_3.text()

        input4=self.lineEdit\_4.text()

        input5=self.lineEdit\_5.text()

        nisab=self.lineEdit\_7.text()

        print(input1,input2,input3,input4,input5)

        if input1 == "":

            input1 = 0

        if input2 == "":

            input2 = 0

        if input3 == "":

            input3 = 0

        if input4 == "":

            input4 = 0

        if input5 == "":

            input5 = 0

        output1=int(input1)+int(input2)+int(input3)+int(input4)-int(input5)

        self.lineEdit\_6.setText(str(output1))

        if output1 > int(nisab):

            output2 = int(output1\*0.025)

            self.lineEdit\_8.setText(str(output2))

        else:

            self.lineEdit\_8.setText("0")

class Uikalkulatorzakatprofesi(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(Uikalkulatorzakatprofesi, self).\_\_init\_\_()

        uic.loadUi('kalkulatorzakatprofesi.ui', self)

        self.setWindowTitle("Kalkulator Zakat profesi")

        self.pushButton.clicked.connect(self.hitung)

* 1. **Takmir.py**

from PyQt5 import QtWidgets, uic, QtSql, QtCore

from PyQt5.QtGui import \*

from PyQt5.QtCore import \*

import sys

from PyQt5.QtSql import QSqlTableModel

import sqlite3

import code

class UiTakmir(QtWidgets.QDialog):

    def \_\_init\_\_(self):

        super(UiTakmir, self).\_\_init\_\_()

        uic.loadUi("infodialog.ui",self)

        self.pushButton\_2.clicked.connect(self.backtomenu)

        self.setWindowModality(Qt.ApplicationModal)

        conn=sqlite3.connect('pemasukan\_kas.db')

        c=conn.cursor()

        ketua=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Ketua Umum'")

        for x in ketua:

            hasil1=x[0]

        self.lineEdit.setText(str(hasil1))

        self.pushButton\_2.clicked.connect(self.zakatmal)

        self.setWindowModality(Qt.ApplicationModal)

        nsb= int((85\*660485)/12)

        self.lineEdit\_7.setText(str(nsb))

    def zakatmal(self):

        self.showzakatmal=Uikalkulatorzakatmal()

        self.close()

        self.showzakatmal.show()

    def hitung(self):

        input1=self.lineEdit\_2.text()

        input2=self.lineEdit\_3.text()

        input3=self.lineEdit\_4.text()

        nisab=self.lineEdit\_7.text()

        print(input1,input2,input3)

        if input1 == "":

            input1 = 0

        if input2 == "":

            input2 = 0

        if input3 == "":

            input3 = 0

        nisab=int(nisab)

        output1=int(input1)+int(input2)-int(input3)

        self.lineEdit\_6.setText(str(output1))

        if output1 > nisab:

            output2 = int(output1\*0.025)

            self.lineEdit\_8.setText(str(output2))

        else:

            self.lineEdit\_8.setText("0")

        wkketua=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Wakil Ketua'")

        for x in wkketua:

            hasil2=x[0]

        self.lineEdit\_7.setText(str(hasil2))

        dewan=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Dewan penasihat'")

        for x in dewan:

            hasil3=x[0]

        self.lineEdit\_4.setText(str(hasil3))

        sekretaris=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Sekretaris'")

        for x in sekretaris:

            hasil4=x[0]

        self.lineEdit\_2.setText(str(hasil4))

        wksekretaris=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Wakil Sekretaris'")

        for x in wksekretaris:

            hasil5=x[0]

        self.lineEdit\_3.setText(str(hasil5))

        bendahara=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Bendahara'")

        for x in bendahara:

            hasil6=x[0]

        self.lineEdit\_5.setText(str(hasil6))

        wkbendahara=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Wakil bendahara'")

        for x in wkbendahara:

            hasil7=x[0]

        self.lineEdit\_6.setText(str(hasil7))

        bpendidikan=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Bidang Pendidikan'")

        for x in bpendidikan:

            hasil8=x[0]

        self.lineEdit\_8.setText(str(hasil8))

        bpembangunan=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Bidang Pembangunan'")

        for x in bpembangunan:

            hasil9=x[0]

        self.lineEdit\_9.setText(str(hasil9))

        bsosial=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Bidang Sosial'")

        for x in bsosial:

            hasil10=x[0]

        self.lineEdit\_10.setText(str(hasil10))

        bperibadatan=c.execute("SELECT Nama FROM pengurus WHERE Jabatan='Bidang Peribadatan'")

        for x in bperibadatan:

            hasil11=x[0]

        self.lineEdit\_11.setText(str(hasil11))

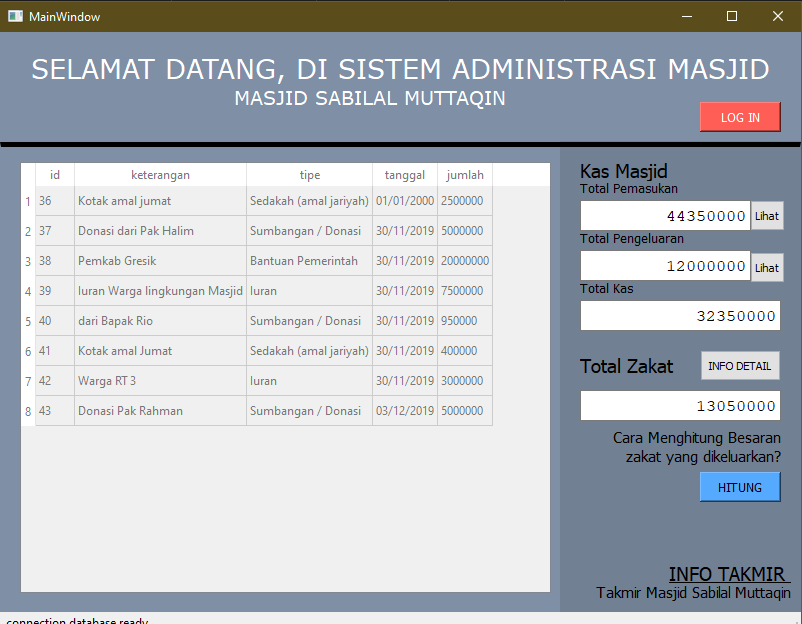
    def backtomenu(self):

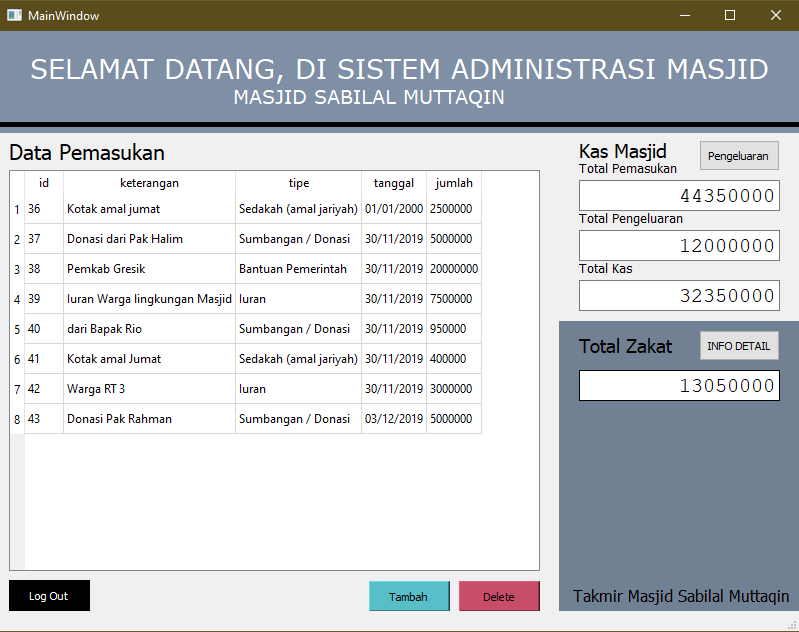
        self.menu=code.Ui()

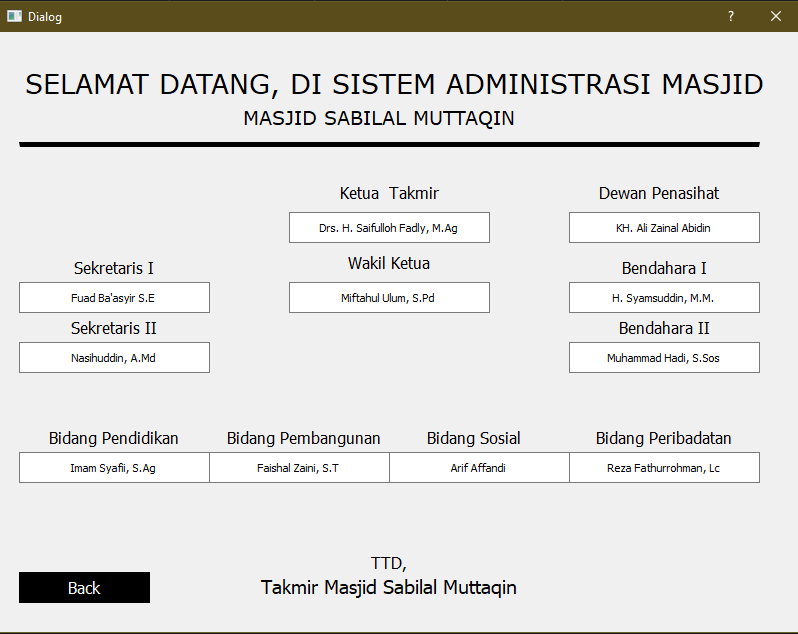
        self.menu.show()

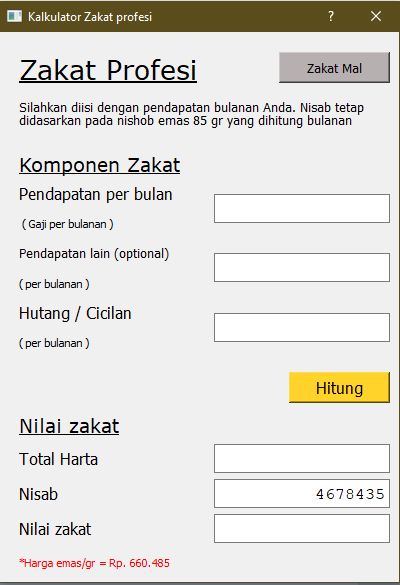
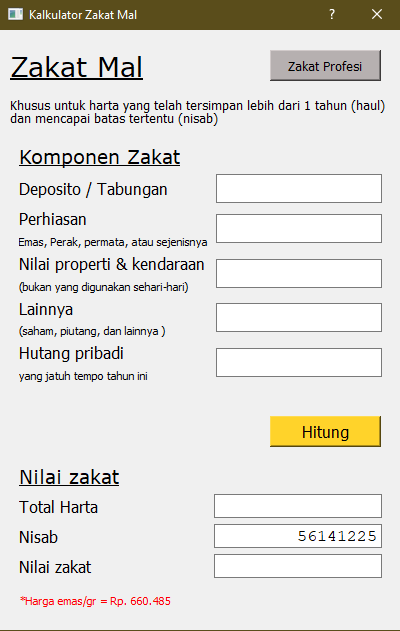
        self.close()

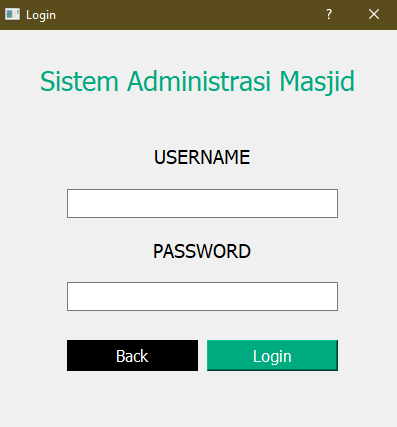
1. Hasil Running Program

Homepage

Homeadmin

Info takmir

Kalkulator zakat

login

Tambah data

