**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

**Кафедра «Направляющие телекоммуникационные среды»**

**Отчет по лабораторной работе №8**

по дисциплине «Введение в информационные технологии» на тему:

Создание визуального интерфейса для базы данных

Выполнил:

Кононов Г.А.  
Группа: БИН 2301

Москва 2023

**Цель работы:** Создать визуальный интерфейс для базы данных

Визуальный интерфейс основан на представлении состояния системы в виде некоторого набора визуальных образов на устройстве отображения. Каждый из компонентов системы, с которым пользователь может работать, отображается в виде значка или надписи, расположенных в определенном месте экрана. Особенности отображения этого значка или надписи могут служить индикатором состояния данного компонента системы.

**Техническое задание:**

Создать оконное приложение позволяющее редактировать базу данных с расписанием Вашей группы.

**Минимальные требования к разрабатываемой системе:**

1. Использование библиотеки PyQt5.

2. Использование адаптера psycopg2.

3. Приложение должно иметь при себе функционал, позволяющий: просматривать базу данных в удобном для пользователя формате, удалять, добавлять и изменять записи в этой же базе данных.

4. Визуальная часть должна иметь при себе:

a. Минимум 3 вкладки, в каждой из которых содержится информация из отдельной таблицы в базе данных.

b. Внутри каждой вкладки информация должна отображаться в виде таблиц.

c. Внутри каждой вкладки должна отображаться кнопка с обновлением информации.

d. Внутри каждой таблицы должны отображаться все поля из таблицы в базе данных в виде колонок

e. Внутри каждой таблицы после каждой строки записи должны быть отображены кнопки изменения и удаления записи

f. В конце каждой таблицы должна находиться пустая строка с кнопкой для добавления новой записи.

g. На вкладке с расписанием дни недели должны быть указаны в отдельных таблицах.

import psycopg2

import sys

from PyQt5.QtWidgets import (QApplication, QWidget,QTabWidget, QAbstractScrollArea,QVBoxLayout, QHBoxLayout,QTableWidget, QGroupBox,QTableWidgetItem, QPushButton, QMessageBox)

class MainWindow(QWidget):

daysOfWeek=["Понедельник","Вторник","Среда","Четверг","Пятница","Суббота"]

alltabs={}

def \_\_init\_\_(self):

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

self.\_connect\_to\_db()

self.setWindowTitle("Shedule")

self.vbox = QVBoxLayout(self)

self.tabs = QTabWidget(self)

self.vbox.addWidget(self.tabs)

for day in self.daysOfWeek:

self.\_create\_shedule\_tab(day)

def \_connect\_to\_db(self):

self.conn = psycopg2.connect(database="bin2108",

user="postgres",

password="1234",

host="localhost",

port="5432")

self.cursor = self.conn.cursor()

def \_create\_shedule\_tab(self,day):

self.shedule\_tab = QWidget()

self.tabs.addTab(self.shedule\_tab, day)

self.monday\_gbox = QGroupBox(day)

self.svbox = QVBoxLayout()

self.shbox1 = QHBoxLayout()

self.shbox2 = QHBoxLayout()

self.svbox.addLayout(self.shbox1)

self.svbox.addLayout(self.shbox2)

self.shbox1.addWidget(self.monday\_gbox)

self.\_create\_table(day)

self.update\_shedule\_button = QPushButton("Update")

self.shbox2.addWidget(self.update\_shedule\_button)

self.update\_shedule\_button.clicked.connect(self.\_update\_shedule)

self.shedule\_tab.setLayout(self.svbox)

def \_create\_table(self,day):

self.table = QTableWidget()

self.table.setSizeAdjustPolicy(QAbstractScrollArea.AdjustToContents)

self.table.setColumnCount(4)

self.table.setHorizontalHeaderLabels(["Чётность", "Время", "Предмет",""])

self.\_update\_table(self.table,day)

self.mvbox = QVBoxLayout()

self.mvbox.addWidget(self.table)

self.monday\_gbox.setLayout(self.mvbox)

self.alltabs[day]=self.table

def \_update\_table(self,table:QTableWidgetItem,day):

self.cursor.execute("SELECT \* FROM timetable WHERE day=%s;",(day,))

records = list(self.cursor.fetchall())

table.setRowCount(len(records)+1)

table.setSortingEnabled(True)

self.c=0

for i, r in enumerate(records):

r = list(r)

self.c+=1

joinButton = QPushButton("Обновить")

table.setItem(i, 0,

QTableWidgetItem(str(r[1])))

table.setItem(i, 1,

QTableWidgetItem(str(r[2])))

table.setItem(i, 2,

QTableWidgetItem(str(r[3])))

table.setCellWidget(i, 3, joinButton)

joinButton.clicked.connect(lambda \*,t=table,num=i,d=day:self.\_change\_day\_from\_table(t,num,d))

joinButton = QPushButton("Добавить")

table.setItem(self.c, 0,QTableWidgetItem(""))

table.setItem(self.c, 1,QTableWidgetItem(""))

table.setItem(self.c, 2,QTableWidgetItem(""))

table.setCellWidget(self.c, 3, joinButton)

joinButton.clicked.connect(lambda \*,t=table,num=-1,d=day:self.\_change\_day\_from\_table(t,num,d))

table.resizeRowsToContents()

def \_change\_day\_from\_table(self,table, rowNum,day):

row = list()

for i in range(table.columnCount()):

try:

row.append(table.item(rowNum, i).text())

except:

row.append(None)

try:

if rowNum==-1:

self.cursor.execute("INSERT INTO timetable (text,day,parity,time) VALUES(%s,%s,%s,%s);", (row[2],day,row[0],row[1]))

else:

self.cursor.execute("UPDATE timetable SET text=%s WHERE day=%s AND parity=%s AND time=%s;", (row[2],day,row[0],row[1]))

self.conn.commit()

except:

row.append(None)

try:

if rowNum==-1:

self.cursor.execute("INSERT INTO timetable (text,day,parity,time) VALUES(%s,%s,%s,%s);", (row[2],day,row[0],row[1]))

else:

self.cursor.execute("UPDATE timetable SET text=%s WHERE day=%s AND parity=%s AND time=%s;", (row[2],day,row[0],row[1]))

self.conn.commit()

except:

QMessageBox.about(self, "Error", "Enter all fields")

def \_update\_shedule(self):

for day,table in self.alltabs.items():

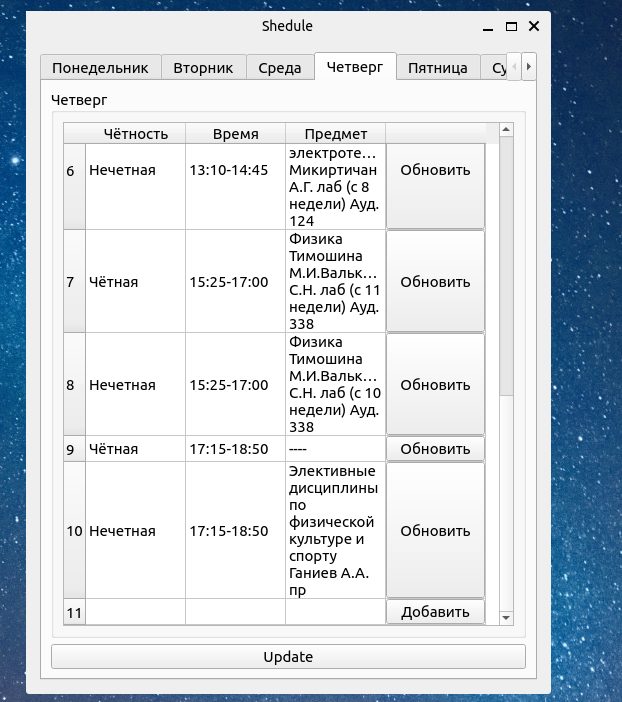
self.\_update\_table(table,day)

app = QApplication(sys.argv)

win = MainWindow()

win.show()

sys.exit(app.exec\_())



**Рисунок 1 – Работа приложения**

**Вывод:** Приложения с графическим интерфейсом стали неотъемлемой частью нашей жизни. Для работы с консольными приложениями требуется некоторые навыки, в то время как в графических это в разы проще. В этой лабораторной работе был создано оконное приложение для расписания своей группы, в котором можно его отредактировать