**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Отчет

По лабораторной работе №3

На тему: “ Исследование способов построения интерфейса пользователя на базе QtWidgets”

По дисциплине: “Кроссплатформенное программирование”

Выполнил:

ст.гр. ИС/б-17-2

Черняев Н.Г.

Проверил:

Строганов В.А.

Севастополь

2021

1 ЦЕЛЬ РАБОТЫ

Изучить основные методики создания графического пользовательского интерфейса с использованием виджетов QtWidgets. Приобрести навыки разработки интерфейса пользователя для приложений на основе фреймворка Qt.

2 ПОСТАНОВКА ЗАДАЧИ

2.1. Создать проект Qt Gui Application.

2.2. Разработать графический интерфейс приложения согласно варианту задания.

2.3. Используя механизм сигналов и слотов добавить функционал, согласно варианту задания.

Вариант 5

Тип формы – Форма тестирования знаний (3 вопроса, содержание название (Line Edit) и варианты ответа (Check Box, Radio Button, Combo Box)).

Функционал – Валидация (Наличие текста в полях, соответствие типам вводимых данных).

Метод разработки – Qt designer

3 ТЕКСТ ПРОГРАММЫ

mainwindow.h:

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include <QMessageBox>

QT\_BEGIN\_NAMESPACE

namespace **Ui** { class **MainWindow**; }

QT\_END\_NAMESPACE

class **MainWindow** : public QMainWindow

{

Q\_OBJECT

public:

**MainWindow**(QWidget \*parent = nullptr);

~***MainWindow***();

private:

Ui::MainWindow \*ui;

private slots:

void **validate**();

};

#endif // MAINWINDOW\_H

mainwindow.cpp:

#include "mainwindow.h"

#include "ui\_mainwindow.h"

MainWindow::**MainWindow**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

connect(ui->pushButton, SIGNAL(clicked()), this, SLOT(validate()));

}

MainWindow::~***MainWindow***()

{

delete ui;

}

void MainWindow::**validate**()

{

QString userName = ui->lineEdit->text();

bool radio1 = ui->radioButton->isChecked();

bool radio2 = ui->radioButton\_2->isChecked();

bool nameContainsNumber = !QRegExp("\\D\*").exactMatch(userName);

bool radioIsNotChecked = !(radio1 || radio2);

QMessageBox msgBox;

if (userName == "" || radioIsNotChecked || nameContainsNumber)

{

QString errorsText = "";

if (userName == "") errorsText += "Заполните ваше имя! \n";

if (userName != "" && nameContainsNumber) errorsText += "Имя содержит недопустимые символы! \n";

if (radioIsNotChecked) errorsText += "Вы не ответили на второй вопрос";

msgBox.setWindowTitle("Ошибка");

msgBox.setText(errorsText);

}

else

{

msgBox.setWindowTitle("Удача");

msgBox.setText("Ваши ответы были отправлены!");

}

msgBox.setStandardButtons(QMessageBox::Ok);

msgBox.*exec*();

}

4 РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Ниже представлен интерфейс приложения, который видит пользователь при первом открытии программы.

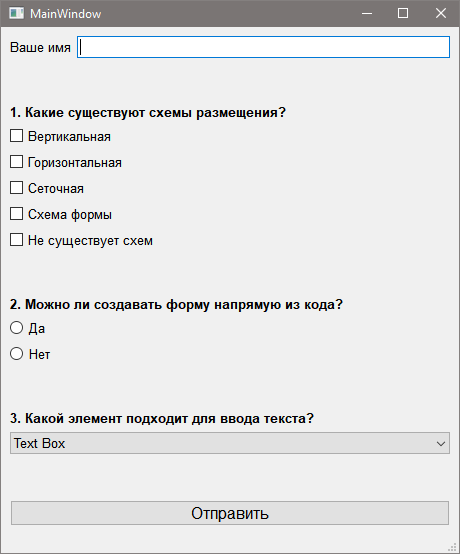


Рисунок 1 – Старт программы

Если мы не заполним поля и нажмем кнопку отправки формы, то откроется диалоговое окно с соответствующим сообщением.

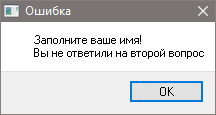


Рисунок 2 – Диалоговое окно при отправке некорректной формы

Если в поле “Ваше имя” ввести что-либо кроме букв, то также появится ошибка с сообщением о недопустимых символах. Данное окно представлено на рисунке 3.

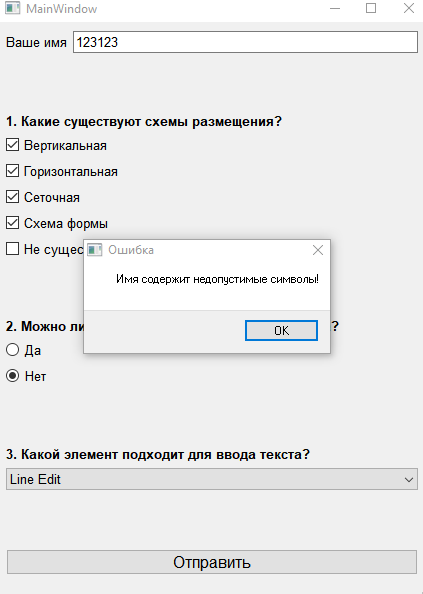


Рисунок 3 – Проверка элемента Line Edit на тип вводимых данных

Если при нажатии кнопки отправления все поля удовлетворяют условию, то будет показано окно с сообщением об успешной отправке данных (рисунок 4)

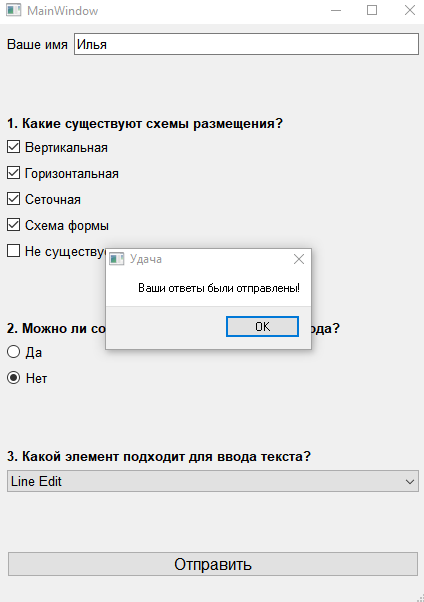


Рисунок 4 – Отправка формы при верно введенных данных

ВЫВОДЫ

В ходе выполнения лабораторной работы были изучены базовые элементы пользовательского интерфейса – QtWidget. При помощи вертикальной и горизонтальной схемы размещения был сделан растягивающийся интерфейс.