МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт информационных технологий и управления в технических системах

Кафедра «Информационные системы»

**ОТЧЕТ**

о выполнении лабораторной работы № 4

по дисциплине “Кроссплатформенное Программирование”

Выполнил: ст. гр. ИС/б-20-2-о

Белик Г. М.

Проверил: доц. каф.

«Информационные системы»

Строганов В. А.

Севастополь

2023

**ЛАБОРАТОРНАЯ РАБОТА №4**

**«Исследование способов построения интерфейса пользоваеля с помощью языка разметки QML»**

**Цель работы**

Изучить основы языка разметки QML. Приобрести практические навыки создания графических интерфейсов Qt-приложений на основе разметки.

**Задачи**

1. Выполнить проектирование графического интерфейса приложения, согласно варианту задания.
2. Создать проект Qt Quick 2 приложение.
3. Добавить собственный класс Button в проект.
4. Создать в дизайнере интерфейс, согласно варианту задания.
5. Реализовать логику приложения средствами QML по варианту.
6. Выполнить сравнительное исследование методов построения интерфейса пользователя, рассмотренных в лабораторной работе №3, и на основе QML. Сравнение провести по критериям: 1) трудоемкости реализации; 2) гибкости получаемого программного решения.

**Вариант 1**

При нажатии на кнопку «Сравнить», осуществить сравнение текста в первом поле ввода с текстом во втором. Результат сравнения вывести соответствующим сообщением в текстовую метку («Текст совпадает» или «Текст не совпадает»)

**Ход работы**

Ниже представлен интерфейс приложения, который видит пользователь при запуске программы (Рисунок 1). Также показан пример отображения в случае изменения размеров окна пользователем (Рисунок 2).

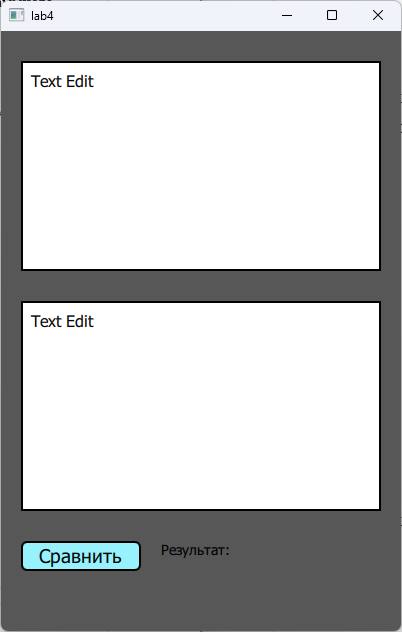


Рисунок 1 – Запуск программы

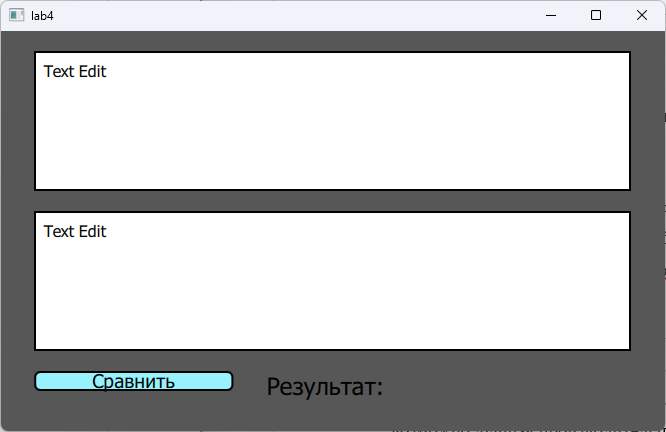
****

Рисунок 2 – Старт програмы с измененным размером окна

После нажания кнопки “Сравнить” необходимо получить результат сравнения, для начала, введем одинковое содержимое (Рисунок 3), а также не совпадающий текст (Рисунок 4).

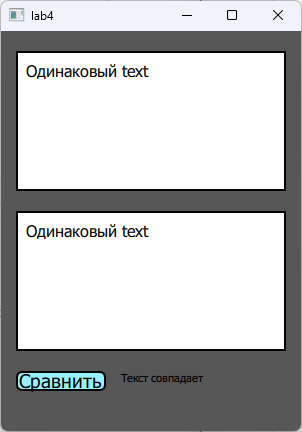


Рисунок 3 – Результат в случае совпадения текста

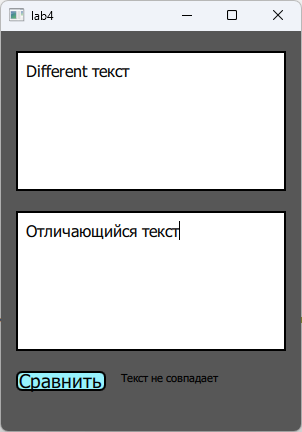


Рисунок 4 – Результат в случае различного текста

**Текст программы:**

**Button.qml**

import QtQuick 2.0

Rectangle {

id: *button*

property int buttonHeight: 75

property int buttonWidth: 150

property string label:"Сравнить"

property real labelSize: 14

radius: 6

antialiasing: true

border {

width: 2

color: "#000"

}

width: *buttonWidth*;

height: *buttonHeight*

color: "#97f2fd"

Text {

id: *buttonLabel*

anchors.centerIn: *parent*

text: *label*

color: "#000"

font.pointSize: *labelSize*

Behavior on color {

ColorAnimation {

duration: 200

}

}

}

signal buttonClick()

MouseArea {

id: *buttonMouseArea*

anchors.fill: *parent*

onClicked: *buttonClick*()

hoverEnabled: true

onEntered: {

*parent*.color = "#0857c0"

*buttonLabel*.color = "#fff"

}

onExited: {

*parent*.color = "#97f2fd"

*buttonLabel*.color = "#000"

}

}

scale: *buttonMouseArea*.pressed ? 1.2 : 1

Behavior on scale {

NumberAnimation {

duration: 200

}

}

Behavior on color {

ColorAnimation {

duration: 200

}

}

}

**main.qml**

import QtQuick 2.12

import QtQuick.Window 2.12

Window {

id: *window*

width: 400

height: 600

color: "#575757"

visible: true

title: *qsTr*("lab4")

minimumWidth: 300

minimumHeight: 400

Rectangle {

id: *rectangle*

y: *parent*.height\*0.05

height: *parent*.height\*0.35

width: *parent*.width\*0.9

anchors.horizontalCenter: *parent*.horizontalCenter

color: "#ffffff"

border.width: 2

TextEdit {

id: *textEdit*

width: *parent*.width

height: *parent*.height

anchors.horizontalCenter: *parent*.horizontalCenter

color: "#000000"

text: *qsTr*("Text Edit")

font.pixelSize: 16

selectionColor: "#000000"

selectByMouse: true

textMargin: 10

}

}

Rectangle {

id: *rectangle1*

y: *parent*.height\*0.45

height: *parent*.height\*0.35

width: *parent*.width\*0.9

anchors.horizontalCenter: *parent*.horizontalCenter

color: "#ffffff"

border.width: 2

TextEdit {

id: *textEdit1*

width: *parent*.width

height: *parent*.height

anchors.horizontalCenter: *parent*.horizontalCenter

color: "#000000"

text: *qsTr*("Text Edit")

font.pixelSize: 16

selectionColor: "#000000"

selectByMouse: true

textMargin: 10

}

}

Button {

id: *button*

y: *parent*.height\*0.85

height: *parent*.height\*0.05

width: *parent*.width\*0.3

anchors.left: *parent*.left

anchors.leftMargin: *parent*.width/20

label: "Сравнить"

onButtonClick:

{

*text1*.text = *textEdit*.text == *textEdit1*.text ? "Текст совпадает" : "Текст не совпадает"

}

}

Text {

id: *text1*

y: *parent*.height\*0.85

height: *parent*.height\*0.05

text: *qsTr*("Результат:")

anchors.left: *button*.right

font.pixelSize: *parent*.width/28

anchors.leftMargin: *parent*.width/20

}

}

**Выводы**

В ходе лабораторной работы были исследованы построения интерфейса пользователя с поомщью языка разметки QML. Основной концепцией QML явлется Model–View, что позволяет разделить логику и оторажение и ускоряет процесс адаптации приложения под различные цели. В сравнении с QWidjets данный метод является более адаптивным за счет отделения логики и более гибком для представлений, менее трудоемким в перспективе разработки интерфейса под множества различных устройств.