**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Отчет

По лабораторной работе №4

На тему: “Исследование способов построения интерфейса пользователя с помощью языка разметки QML”

По дисциплине: “Кроссплатформенное программирование”

Выполнил:

ст.гр. ИС/б-17-2

Черняев Н.Г.

Проверил:

Строганов В.А.

Севастополь

2021

1 ЦЕЛЬ РАБОТЫ

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

2 ПОСТАНОВКА ЗАДАЧИ

2.1. Выполнить проектирование графического интерфейса приложения, согласно варианту задания.

2.2. Создать проект Qt Quick 2 приложение.

2.3. Добавить собственный класс Button в проект.

2.4. Создать в дизайнере интерфейс, согласно варианту задания.

2.5. Реализовать логику приложения средствами QML по варианту.

2.6. 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»)

3 ТЕКСТ ПРОГРАММЫ

Button.qml:

import QtQuick 2.0

Rectangle {

id: *button*

property int buttonHeight: 75

property int buttonWidth: 150

property string label

property real labelSize: 14

radius: 3

antialiasing: true

border {

width: 2

color: "#000"

}

width: *buttonWidth*;

height: *buttonHeight*

color: "#fff"

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 = "#000"

*buttonLabel*.color = "#fff"

}

onExited: {

*parent*.color = "#fff"

*buttonLabel*.color = "#000"

}

}

scale: *buttonMouseArea*.pressed ? 1.03 : 1.00

Behavior on scale {

NumberAnimation {

duration: 100

}

}

Behavior on color {

ColorAnimation {

duration: 200

}

}

}

main.qml:

import QtQuick 2.12

import QtQuick.Window 2.12

Window {

id: *window*

width: 350

height: 420

visible: true

color: "#ffffff"

title: *qsTr*("Hello World")

TextEdit {

id: *textEdit*

y: 20

height: 108

color: "#000000"

text: *qsTr*("Text Edit")

anchors.left: *parent*.left

anchors.right: *parent*.right

font.pixelSize: 14

horizontalAlignment: Text.AlignLeft

wrapMode: Text.WordWrap

anchors.rightMargin: 24

anchors.leftMargin: 27

selectionColor: "#000000"

selectByMouse: true

textMargin: 10

padding: 0

leftPadding: 0

topPadding: 0

Rectangle {

id: *rectangle*

y: 0

height: 108

color: "#00ffffff"

border.width: 2

anchors.left: *parent*.left

anchors.right: *parent*.right

anchors.rightMargin: 0

anchors.leftMargin: 0

}

}

TextEdit {

id: *textEdit1*

y: 162

height: 108

color: "#000000"

text: *qsTr*("Text Edit")

anchors.left: *parent*.left

anchors.right: *parent*.right

font.pixelSize: 14

horizontalAlignment: Text.AlignLeft

wrapMode: Text.WordWrap

anchors.rightMargin: 24

anchors.leftMargin: 27

selectionColor: "#000000"

selectByMouse: true

padding: 0

topPadding: 0

leftPadding: 0

textMargin: 10

Rectangle {

id: *rectangle1*

height: 108

color: "#00ffffff"

border.width: 2

anchors.left: *parent*.left

anchors.right: *parent*.right

anchors.rightMargin: 0

anchors.leftMargin: 0

}

}

Button {

id: *button*

y: 307

height: 35

anchors.left: *parent*.left

anchors.right: *parent*.right

anchors.rightMargin: 24

anchors.leftMargin: 27

label: "Сравнить"

onButtonClick:

{

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

}

}

Text {

id: *text1*

y: 361

height: 36

text: *qsTr*("Введите текст и нажмите кнопку")

anchors.left: *parent*.left

anchors.right: *parent*.right

font.pixelSize: 16

horizontalAlignment: Text.AlignHCenter

verticalAlignment: Text.AlignVCenter

wrapMode: Text.WordWrap

anchors.rightMargin: 24

anchors.leftMargin: 27

}

}

4 РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Введем одинаковый текст в поля ввода и получим результат “Текст совпадает”. Результат представлен на рисунке 1.

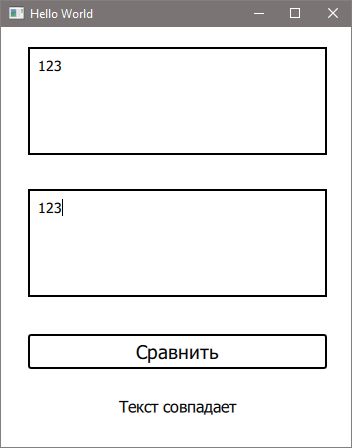


Рисунок 1 – Вывод результата при совпадении текста

Введем разный текст и получим сообщение “Текст не совпадает” (рисунок 2).

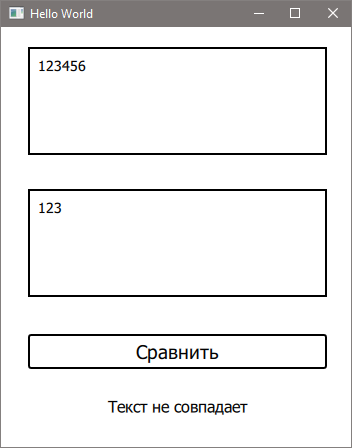


Рисунок 2 – Вывод результата при не совпадении текста

ВЫВОДЫ

В ходе выполнения лабораторной работы были изучены основы языка разметки QML. Выделены основные отличия QtWidget от элементов, описанных при помощи языка QML:

* гибкость при использовании QML
* удобный синтаксис при использовании QML, т.к. он основан на языке JS
* QML в основном ориентирован на мобильный устройства
* возможно лучшая производительность у QtWidgets