**Лабораторная работа №4**

**«ИССЛЕДОВАНИЕ СПОСОБОВ ПОСТРОЕНИЯ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ С ПОМОЩЬЮ ЯЗЫКА РАЗМЕТКИ QML»**

**Цель работы**

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

**Постановка задачи**

1. Изучить принципы построения приложений на основе шаблона MVC (выполняется в ходе самостоятельной подготовки к лабораторной работе).

2. Выполнить проектирование графического интерфейса приложения, согласно варианту задания (выполняется в ходе самостоятельной подготовки к лабораторной работе).

3. Создать проект Qt Quick 2 приложение.

4. Добавить собственный класс Button в проект. Изменить его поведение под свой вариант, если необходимо.

5. Создать в дизайнере интерфейс, согласно варианту задания.

6. Реализовать логику приложения средствами QML по варианту.

7. 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.

**Ход работы**

1. Был создан графический интерфейс приложения, согласно варианту задания.

а

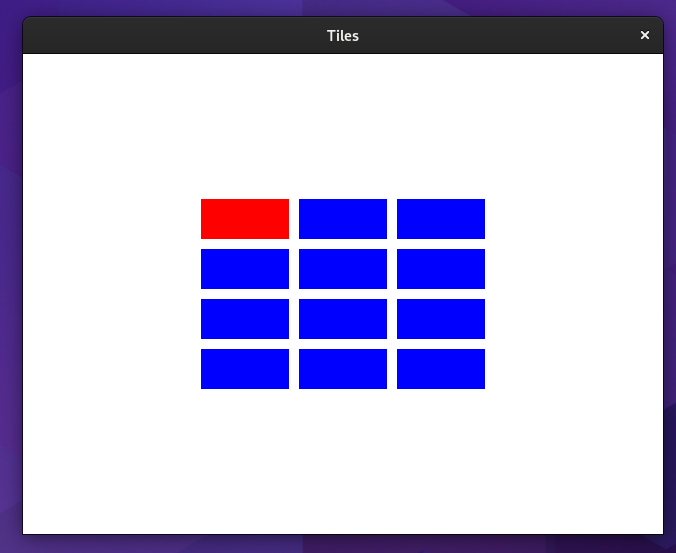


Рисунок 1 – Графический интерфейс приложения

2. Была реализована логика приложения средствами QML, при нажатии на красную кнопку, её цвет меняется на синий, а цвет другой случайно выбранной кнопки меняется на красный.

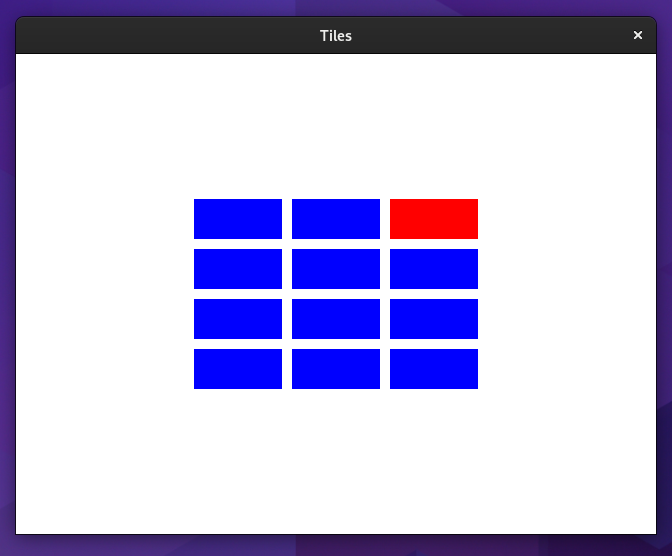


Рисунок 2 – Логика работы приложения

**Выводы**

В ходе выполнения лабораторной работы были изучены основы языка разметки QML. Приобретены практические навыки создания графических интерфейсов Qt-приложений на основе разметки.

Разработка при помощи QML занимает больше времени, чем более старый подход, рассмотренный в лабораторной работе №3, но гибкость такого подхода значительно выше, так как логика, имеющая отношение к внешнему виду и поведению самого интерфейса отделена от основной программы, что означает, что код основной программы можно свободно и независимо от интерфейса изменять.

**ПРИЛОЖЕНИЕ А**

**«ИСХОДНЫЙ КОД ПРОГРАММЫ»**

Листинг А1 – Файл main.qml

import QtQuick

import QtQuick.Controls 2.5

Window {

width: 640

height: 480

visible: true

title: qsTr("Tiles")

property int setColor: 1

function buttonClicked(id)

{

if (id === setColor) {

setColor = Math.floor(Math.random() \* (12-1+1)+1);

}

}

Grid {

columns: 3

rows: 4

anchors.horizontalCenter: parent.horizontalCenter

anchors.verticalCenter: parent.verticalCenter

columnSpacing: 10

rowSpacing: 10

Button {

id: but1

height: 40

width: 88

background: Rectangle

{

color: setColor === 1 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(1);

}

}

Button {

id: but2

height: 40

width: 88

background: Rectangle

{

color: setColor === 2 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(2);

}

}

Button {

id: but3

height: 40

width: 88

background: Rectangle

{

color: setColor === 3 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(3);

}

}

Button {

id: but4

height: 40

width: 88

background: Rectangle

{

color: setColor === 4 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(4);

}

}

Button {

id: but5

height: 40

width: 88

background: Rectangle

{

color: setColor === 5 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(5);

}

}

Button {

id: but6

height: 40

width: 88

background: Rectangle

{

color: setColor === 6 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(6);

}

}

Button {

id: but7

height: 40

width: 88

background: Rectangle

{

color: setColor === 7 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(7);

}

}

Button {

id: but8

height: 40

width: 88

background: Rectangle

{

color: setColor === 8 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(8);

}

}

Button {

id: but9

height: 40

width: 88

background: Rectangle

{

color: setColor === 9 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(9);

}

}

Button {

id: but10

height: 40

width: 88

background: Rectangle

{

color: setColor === 10 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(10);

}

}

Button {

id: but11

height: 40

width: 88

background: Rectangle

{

color: setColor === 11 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(11);

}

}

Button {

id: but12

height: 40

width: 88

background: Rectangle

{

color: setColor === 12 ? "#ff0000" : "#0000ff"

}

onClicked: {

buttonClicked(12);

}

}

}

}