
КАФЕДРА

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

Отчет о лабораторной работе №9

Создание двумерного пользовательского интерфейса в среде Qt Creator с
использованием QML

По дисциплине: Проектирование человеко-машинного интерфейса

РАБОТУ ВЫПОЛНИЛИ

СТУДЕНТЫ ГР. №

подпись, дата

инициалы, фамилия

Санкт-Петербург 2024

Цель работы:

Освоение QML для создания пользовательского интерфейса.

Задание:

Разработать и отладить приложение с графическим пользовательским интерфейсом, использующим для описания интерфейса язык QML.

Название и версия используемой среды моделирования:

QT Creator 11.0.3

Based on Qt 6.4.3 (Clang 13.0 (Apple), arm64)

Built on Sep 27 2023 06:47:35

Описание структуры интерфейса:

В данном проекте для создания графического пользовательского интерфейса использован язык QML, который является частью фреймворка Qt. Интерфейс реализован с помощью компонента `ApplicationWindow`, который представляет основное окно приложения. Внутри окна размещены различные элементы управления, которые обеспечивают взаимодействие пользователя с приложением.

1. Основное окно (`ApplicationWindow`):

- **Размеры и заголовок:** Окно имеет размеры 800x600 пикселей и заголовок “Захаров Андрей 4133К”, который отображается в верхней части окна.

2. Элементы управления (`Column`):

- **Button:** Кнопка с текстом “Нажми меня”. При нажатии на кнопку в консоль выводится сообщение “Кнопка нажата!”.
- **CheckBox:** Флажок с текстом “Отметь меня”. При изменении состояния флажка выводится сообщение в консоль о текущем состоянии флажка (отмечен или нет).
- **TextField:** Поле ввода текста с плейсхолдером “Введите текст здесь...”, которое позволяет пользователю ввести текст.
- **Slider:** Ползунок с диапазоном значений от 0 до 100 и начальным значением 50. При изменении значения ползунка в консоль выводится текущее значение.
- **Text:** Текстовый элемент с текстом “Это текстовое поле”, размером шрифта 20 и синим цветом. Этот элемент используется для отображения статического текста.
- **RadioButton:** Радио кнопка с текстом “Выберите меня”. При нажатии на радио кнопку в консоль выводится сообщение “Радио кнопка выбрана”.
- **ComboBox:** Выпадающий список с тремя опциями: “Опция 1”, “Опция 2” и “Опция 3”. При изменении выбранной опции выводится сообщение о текущем индексе выбранной опции.

Эти элементы управления размещены в вертикальном столбце (`Column`), который центрируется внутри родительского контейнера. Интерфейс демонстрирует базовые возможности использования QML для создания интерактивных элементов управления и простого пользовательского интерфейса.

Для реализации интерфейса использованы стандартные компоненты Qt Quick и Qt Quick Controls, которые обеспечивают необходимую функциональность и внешний вид элементов управления.

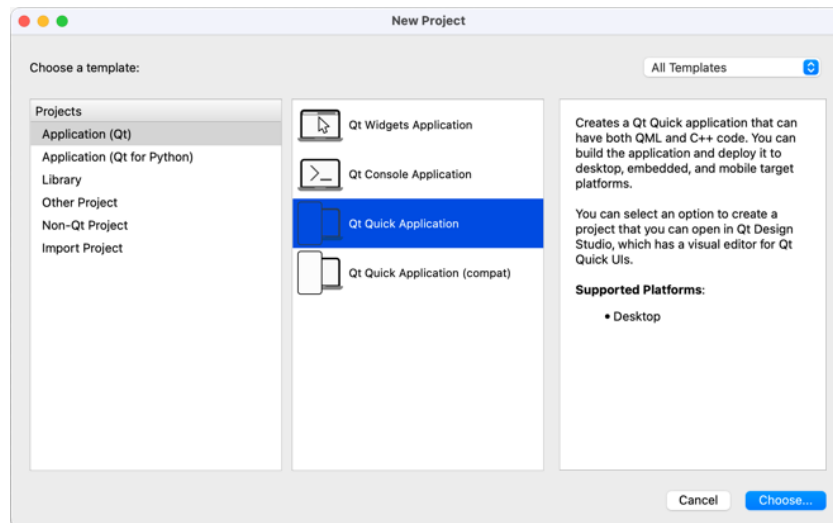


Рисунок 1 – создание нового проекта для QML

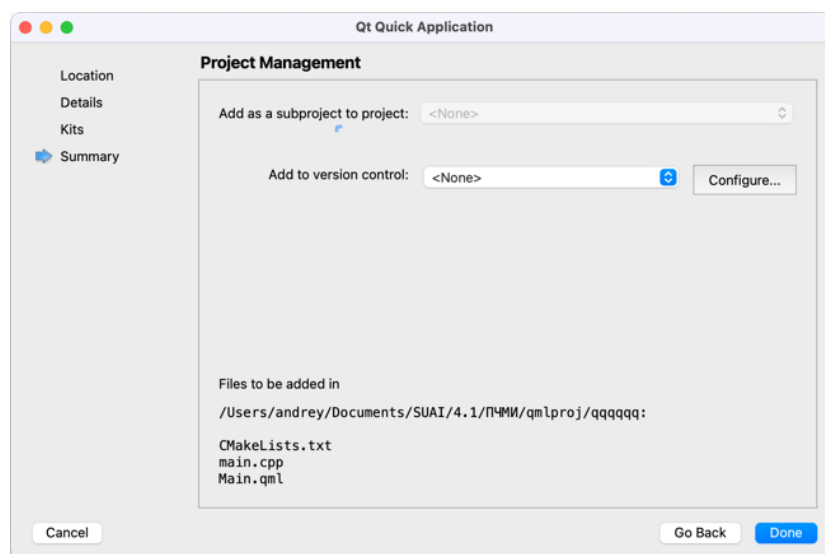


Рисунок 2 – создается qml файл

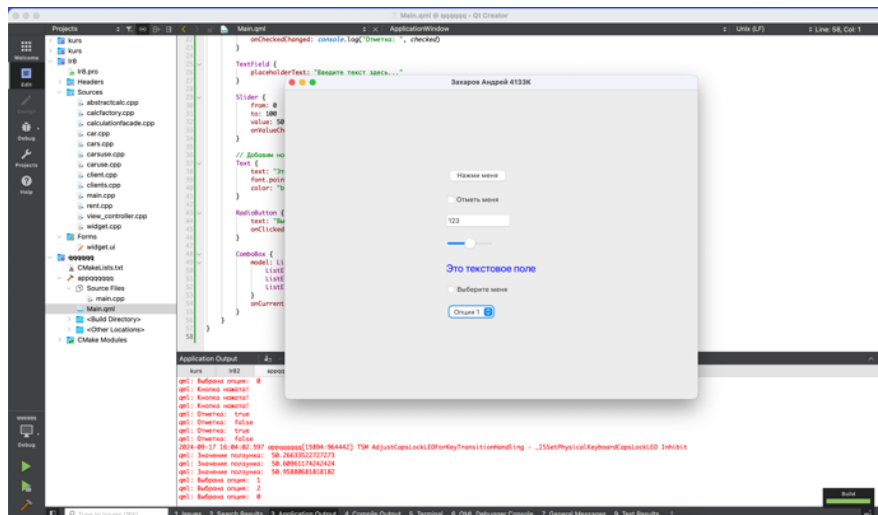


Рисунок 3 – работа приложения

Исходный текст приложения:

Main.cpp:

```
#include <QGuiApplication>
#include <QQmlApplicationEngine>
```

```
int main(int argc, char *argv[])
{
    QGuiApplication app(argc, argv);

    QQmlApplicationEngine engine;
    const QUrl url(u"qrc:/qqqqqq/Main.qml"_qs);
    QObject::connect(&engine, &QQmlApplicationEngine::objectCreationFailed,
        &app, []() { QCoreApplication::exit(-1); },
        Qt::QueuedConnection);
    engine.load(url);

    return app.exec();
}
```

Main.qml:

```
import QtQuick
import QtQuick.Controls 2.15
```

```
ApplicationWindow {
    visible: true
    width: 800
    height: 600
    title: "Захаров Андрей 4133K"
```

```
// Example of using various controls
Column {
    spacing: 20
    anchors.centerIn: parent
```

```

Button {
    text: "Нажми меня"
    onClicked: console.log("Кнопка нажата!")
}

CheckBox {
    text: "Отметь меня"
    onCheckedChanged: console.log("Отметка: ", checked)
}

TextField {
    placeholderText: "Введите текст здесь..."
}

Slider {
    from: 0
    to: 100
    value: 50
    onValueChanged: console.log("Значение ползунка: ", value)
}

// Добавим новый элемент - текстовое поле и флажок
Text {
    text: "Это текстовое поле"
    font.pointSize: 20
    color: "blue"
}

RadioButton {
    text: "Выберите меня"
    onClicked: console.log("Радио кнопка выбрана")
}

ComboBox {
    model: ListModel {
        ListElement { name: "Опция 1" }
        ListElement { name: "Опция 2" }
        ListElement { name: "Опция 3" }
    }
    onCurrentIndexChanged: console.log("Выбрана опция: ", currentIndex)
}
}
}

```

Выводы:

В данной работе был реализован пользовательский интерфейс с использованием языка QML, предоставляющий базовые элементы управления, такие как кнопки, флажки, текстовые поля и ползунки. Интерфейс демонстрирует способность QML создавать интерактивные и функциональные компоненты, обеспечивая простоту и гибкость в разработке графического пользовательского интерфейса.