

Aurora OS глазами Android и iOS разработчика



**Алексей
Гладков**

Mobile Developer

YOUTUBE [mobiledeveloper](#)

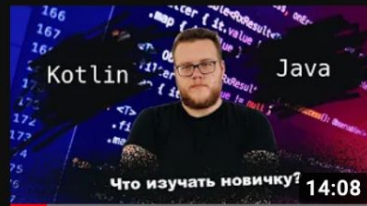
EMAIL mobiledevelopercourse@gmail.com

Алексей Гладков



Обо мне

- Пишу нативные приложения уже 8 лет (Android и iOS)
- Автор канала **Mobile Developer**



Что учить новичку в
Android: Java vs Kotlin?...

47 тыс. просмотров •
11 месяцев назад •



RxJava - Observable,
Flowable. Полный обзор....

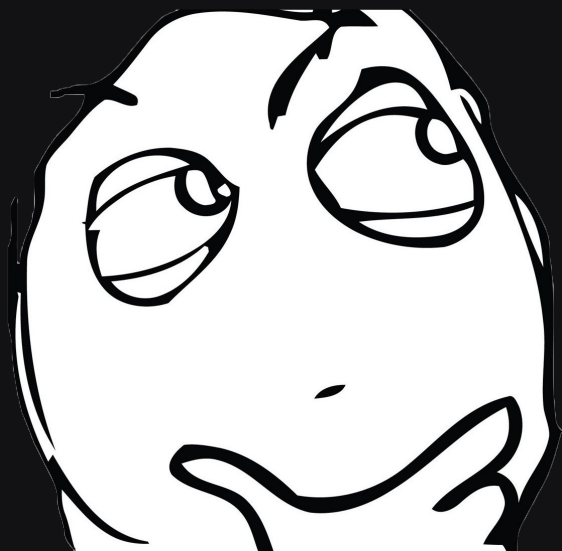
32 тыс. просмотров •
3 года назад •



Dagger 2 - @Provides,
@Module. Полный обзор....

22 тыс. просмотров •
3 года назад •





Какие мобильные ОС вы
знаете?

Предыстория



Предыстория

> Называлась mAemo



2005

Предыстория

- > Называлась mAemo
- > На базе Debian Linux (сильно переработана)



2005

Предыстория

- > Называлась mAemo
- > На базе Debian Linux (сильно переработана)
- > Поддержка Nokia 770, Nokia N800, Nokia N810, Nokia N900



2005

Предыстория

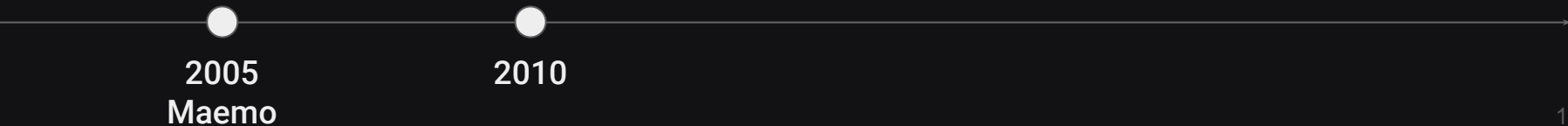
- > Называлась mAemo
- > На базе Debian Linux (сильно переработана)
- > Поддержка Nokia 770, Nokia N800, Nokia N810, Nokia N900
- > Частично открытое ПО



2005

Предыстория

> Называлась MeeGo

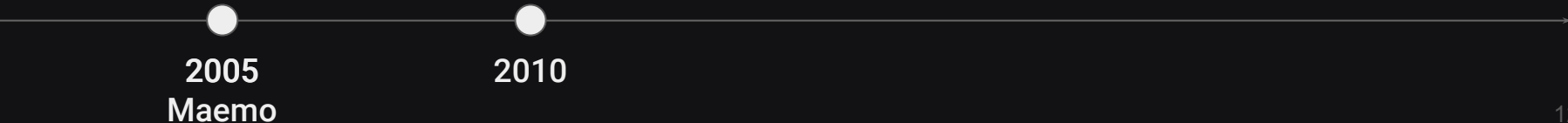


2005
Maemo

2010

Предыстория

- > Называлась MeeGo
- > Компания MeeGo Technical Steering Group



2005
Maemo

2010

Предыстория

- > Называлась MeeGo
- > Компания MeeGo Technical Steering Group
- > Поддержка планшетов, смартфонов, нетбуков

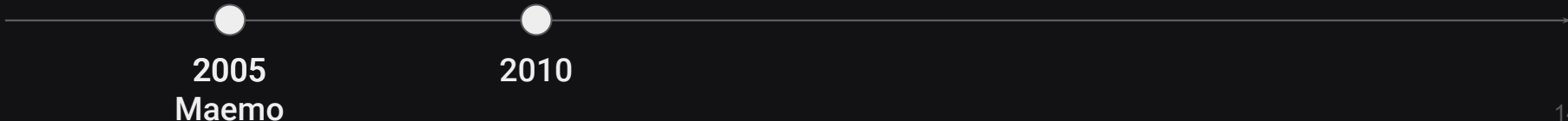


2005
Maemo

2010

Предыстория

- > Называлась MeeGo
- > Компания MeeGo Technical Steering Group
- > Поддержка планшетов, смартфонов, нетбуков
- > Разработка на Qt



2005
Maemo

2010

Предыстория

> Платформа Mer



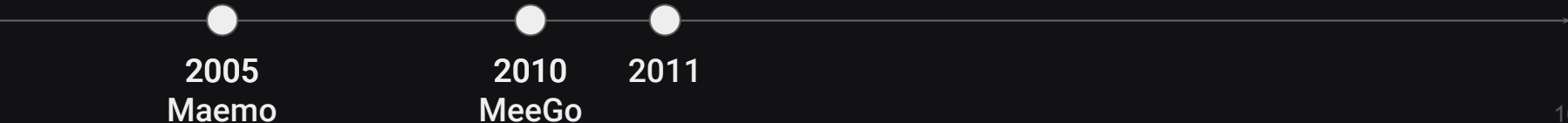
2005
Maemo

2010
MeeGo

2011

Предыстория

- > Платформа Mer
- > Сообщество разработчиков



A horizontal timeline with three white circular markers. The first marker is at 2005 with the label 'Maemo' below it. The second marker is at 2010 with the label 'MeeGo' below it. The third marker is at 2011 with no label below it.

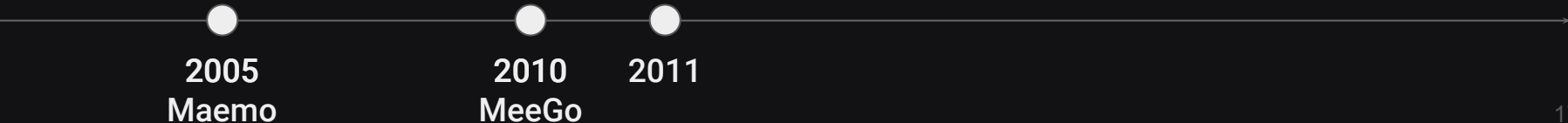
2005
Maemo

2010
MeeGo

2011

Предыстория

- > Платформа Mer
- > Сообщество разработчиков
- > Fork MeeGo



A horizontal timeline with three white circular markers. The first marker is at 2005 with the label 'Maemo' below it. The second marker is at 2010 with the label 'MeeGo' below it. The third marker is at 2011 with no label below it. The timeline ends with an arrow pointing to the right.

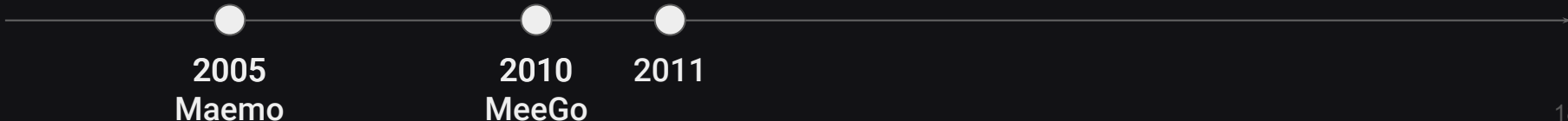
2005
Maemo

2010
MeeGo

2011

Предыстория

- > Платформа Mer
- > Сообщество разработчиков
- > Fork MeeGo
- > Поддержка ARMv6, ARMv7, ARM64, Intel x86, Intel x86-64




2005
Maemo

2010
MeeGo

2011

Предыстория

> Sailfish OS



A horizontal timeline with four white circular markers. Below each marker is a year and a name. The timeline starts at 2005 with Maemo, followed by 2010 with MeeGo, 2011 with Mer, and 2012 with an unlabeled entry.

2005
Maemo


2010
MeeGo

2011
Mer

2012

Предыстория

- > Sailfish OS
- > Компания Jolla (ex-Nokia)



A horizontal timeline with four white circular markers. Below each marker are the year and the name of the operating system.

2005
Maemo


2010
MeeGo

2011
Mer

2012

Предыстория

- > Sailfish OS
- > Компания Jolla (ex-Nokia)
- > Qt Silica View, QtQuick



A horizontal timeline with a light gray line and an arrow pointing to the right. Four white circular markers are placed along the line at regular intervals. Below each marker, the year and the name of the operating system are written in white text.

2005
Maemo

2010
MeeGo

2011
Mer

2012

Предыстория

> Aurora OS



Предыстория

- > Aurora OS
- > Компания Открытая мобильная платформа



Предыстория

- > Aurora OS
- > Компания Открытая мобильная платформа
- > Изменены ядро, загрузчик, системный слой, приложения



Предыстория

- > Aurora OS
- > Компания Открытая мобильная платформа
- > Изменены ядро, загрузчик, системный слой, приложения
- > Основной вектор – безопасность



Предыстория

- > Aurora OS
- > Компания Открытая мобильная платформа
- > Изменены ядро, загрузчик, системный слой, приложения
- > Основной вектор – безопасность
- > Работа в B2B и B2G секторах



Предыстория

- > Aurora OS
- > Компания Открытая мобильная платформа
- > Изменены ядро, загрузчик, системный слой, приложения
- > Основной вектор - безопасность
- > Работа в B2B и B2G секторах
- > Свои UI компоненты для разработки приложений



Предыстория

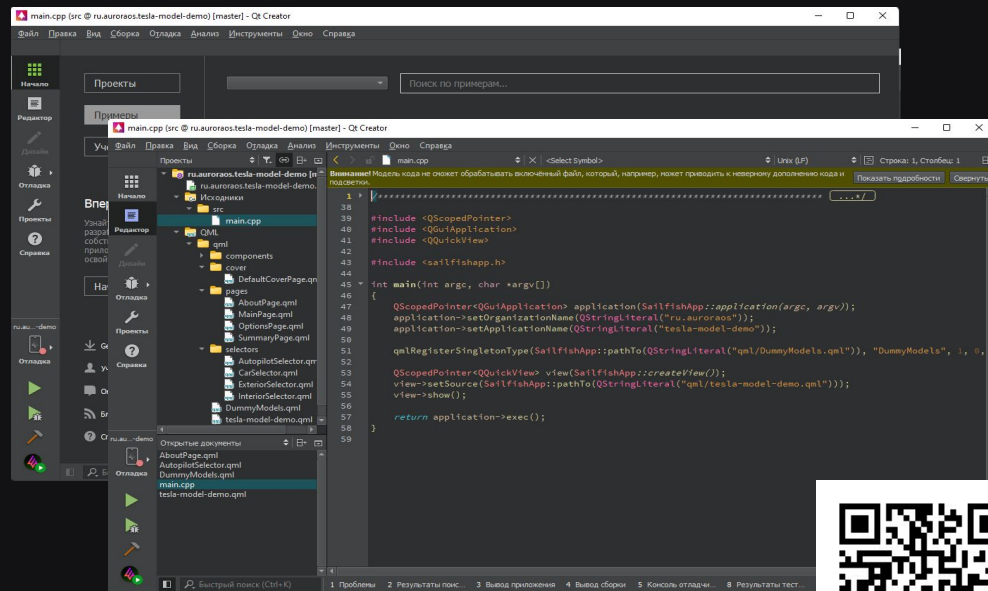
> Aurora OS 4.0



Aurora OS IDE

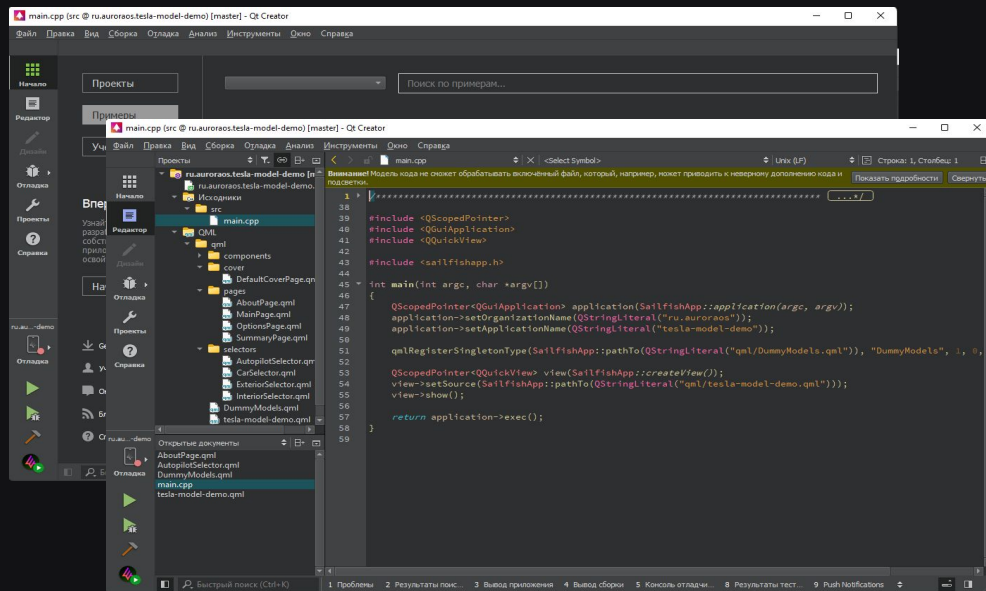
Aurora IDE

- Qt Creator



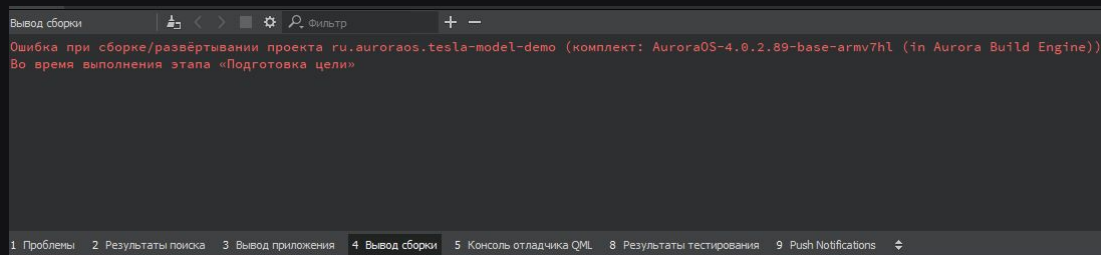
Aurora IDE

- QT Creator
- Невероятно быстрая



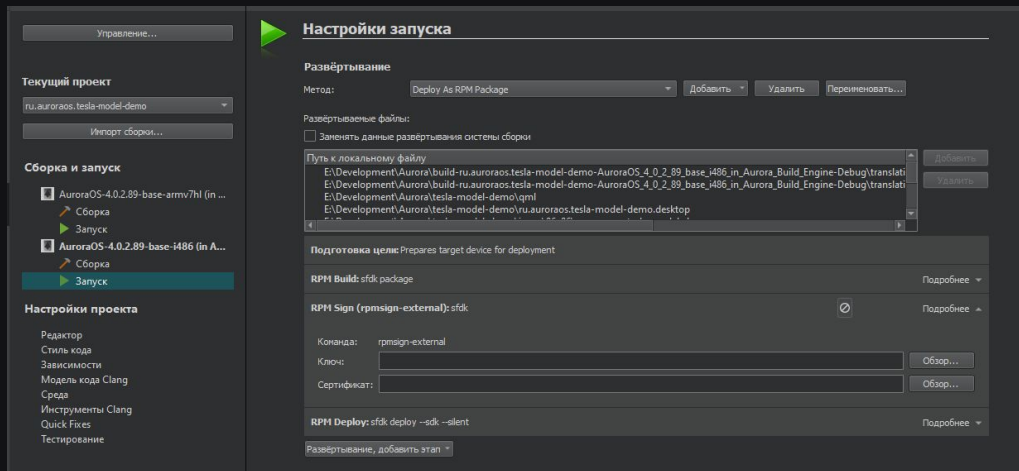
Aurora IDE

- QT Creator
- Невероятно быстрая
- Сложная настройка



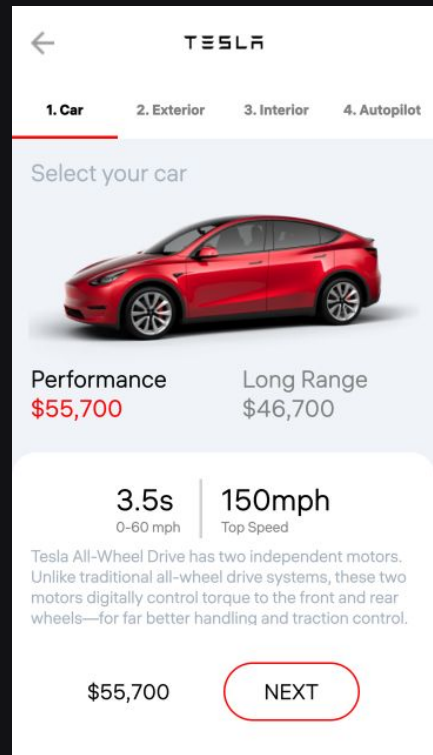
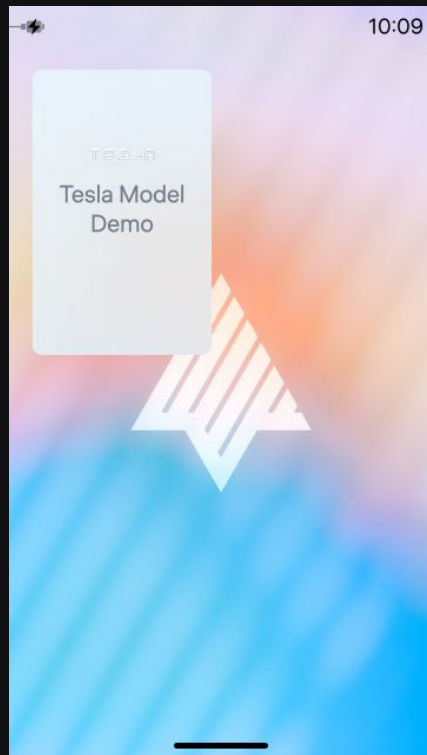
Aurora IDE

- QT Creator
- Невероятно быстрая
- Сложная настройка



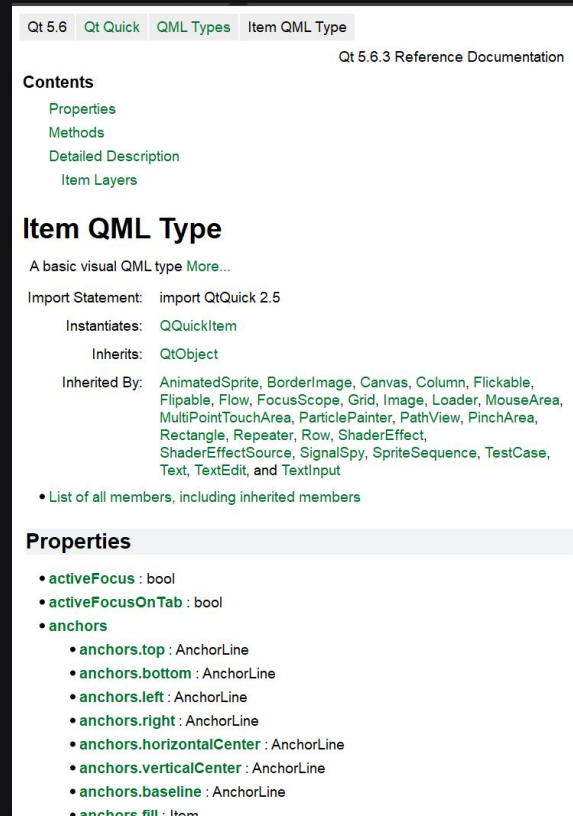
Aurora IDE

- QT Creator
- Невероятно быстрая
- Сложная настройка
- Содержит Emulator (нужен Virtual Box)



Aurora IDE

- QT Creator
- Невероятно быстрая
- Сложная настройка
- Содержит Emulator (нужен Virtual Box)
- Великолепная система помощи



The screenshot shows the Qt 5.6.3 Reference Documentation for the **Item QML Type**. The top navigation bar includes tabs for Qt 5.6, Qt Quick, QML Types, and Item QML Type. The page title is "Qt 5.6.3 Reference Documentation".

Contents

- [Properties](#)
- [Methods](#)
- [Detailed Description](#)
- [Item Layers](#)

Item QML Type

A basic visual QML type [More...](#)

Import Statement: `import QtQuick 2.5`

Instantiates: `QQuickItem`

Inherits: `QObject`

Inherited By: `AnimatedSprite`, `BorderImage`, `Canvas`, `Column`, `Flickable`, `Flipable`, `Flow`, `FocusScope`, `Grid`, `Image`, `Loader`, `MouseArea`, `MultiPointTouchArea`, `ParticlePainter`, `PathView`, `PinchArea`, `Rectangle`, `Repeater`, `Row`, `ShaderEffect`, `ShaderEffectSource`, `SignalSpy`, `SpriteSequence`, `TestCase`, `Text`, `TextEdit`, and `TextInput`

• [List of all members, including inherited members](#)

Properties

- `activeFocus` : `bool`
- `activeFocusOnTab` : `bool`
- `anchors`
 - `anchors.top` : `AnchorLine`
 - `anchors.bottom` : `AnchorLine`
 - `anchors.left` : `AnchorLine`
 - `anchors.right` : `AnchorLine`
 - `anchors.horizontalCenter` : `AnchorLine`
 - `anchors.verticalCenter` : `AnchorLine`
 - `anchors.baseline` : `AnchorLine`
 - `anchors.fill` : `Item`

Документация

Документация

- <https://auroraos.ru/developer>


Главная / Портал разработчика

Мы стремимся развивать сообщество разработчиков приложений под ОС Аврора

Здесь можно найти всю необходимую информацию: обучающие материалы и курсы, документацию, SDK и API.

[Скачать Аврора SDK](#)


Курсы



Разработка программного обеспечения для ОС Аврора

Академия информационных систем | infoystems.ru


49 500 р.




Внедрение, администрирование и работа с платформой управления Аврора Центр

Академия информационных систем | infoystems.ru

30 000 р.



Разработка мобильных приложений из Qt под ОС Аврора



Серия вебинаров о разработке ПО для Аврора



Документация

- <https://auroraos.ru/developer>
- Курс для разработчика InfoSystems
- 49500 рублей

АИС · КУРСЫ · ОСТАВИТЬ ЗАЯВКУ

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ОС АВРОРА (SAILFISH MOBILE OS RUS)

Партнерство
Платформенность

ОСТАВИТЬ ЗАЯВКУ

Фамилия

Укажите Фамилию

Имя

Укажите Имя

Отчество

Укажите Отчество

Название компании

E-mail

Телефон

Телефон

Комментарий

Реквизиты

Дата обучения

21.11.2022 - 23.11.2022

Документация

- <https://auroraos.ru/developer>
- Курс для разработчика InfoSystems
- 49500 рублей
- Курс для разработчика Stepik -
Бесплатно

Разработка мобильных приложений на Qt

Фреймворк Qt, обладая сообществом более, чем из миллиона разработчиков, является одним из самых популярных наборов инструментов для создания мультиплатформенных приложений. Среди аналогов его выделяет высокое быстродействие и обширность предоставляемых API. С 2010 года развиваются функции Qt, направленные на создание...
[Еще](#)

★★★★★ 1
2 259 учащихся

1 отзыв

О курсе

Фреймворк Qt, обладая сообществом более, чем из миллиона разработчиков, является одним из самых популярных наборов инструментов для создания мультиплатформенных приложений. Среди аналогов его выделяет высокое быстродействие и обширность предоставляемых API. С 2010 года развиваются функции Qt, направленные на создание приложений для мобильных устройств: Android, iOS, ОС Аврора/Sailfish OS.

Курс постепенно пополняется, в полном варианте он будет содержать слайды с основными теоретическими материалами и проверочные задания по разработке мобильного ПО с использованием фреймворка Qt.

В ходе курса будут покрыты темы:

1. инструменты, используемые при разработке;
2. работы с общими API: интерфейс пользователя, базы данных, мультимедиа, сеть;
3. использование функций мобильных устройств: датчики, GPS и карты, камера.

Полученные знания позволят выполнить практические задания, которые включают разработку приложений:

1. чат (клиент и сервер) для обмена текстом и файлами;
2. диктофон, проигрыватель мультимедиа, фотокамера;
3. обзор карт с сервиса OSM;
4. список дел.

Материалы подготовлены компанией «Открытая мобильная платформа» при участии преподавателей вузов, в которых проходит обучение разработке ПО для ОС Аврора: МФТИ, ПетрГУ, ЯрГУ, Университет Иннополис.
На данном ресурсе материалы опубликованы по лицензии Creative Commons BY-SA 4.0.

Пожалуйста, сообщите об использовании материалов вне данного курса.

Бесплатно

Поступить на курс

♥ Хочу пройти

Учиться можно сразу

В курс входят

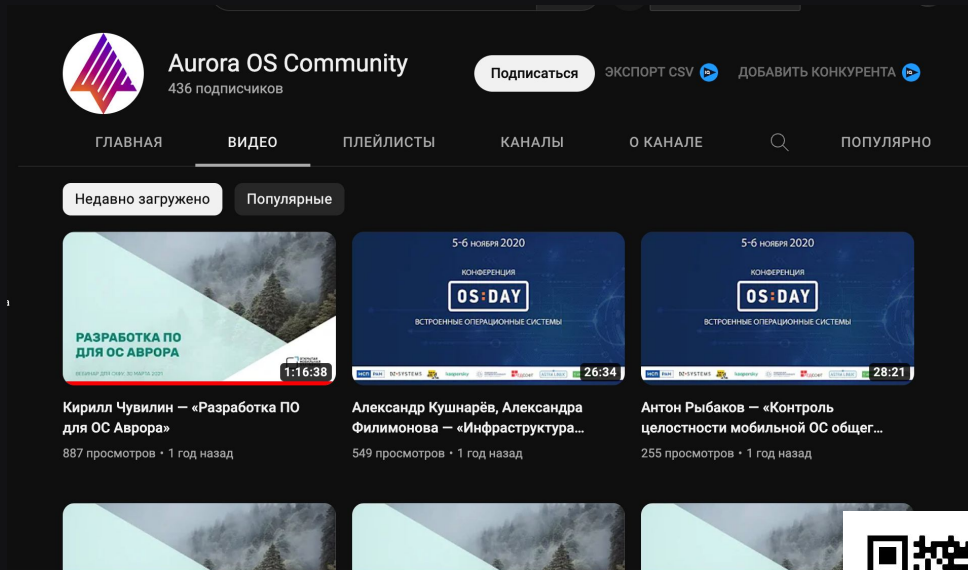
31 урок
37 тестов
5 интерактивных задач

Программа курса

Последнее обновление 16.03.2021

Документация

- <https://auroraos.ru/developer>
- Курс для разработчика InfoSystems
- 49500 рублей
- Курс для разработчика Stepik -
Бесплатно
- Youtube канал aurora os community



Документация

- <https://auroraos.ru/developer>
- Курс для разработчика InfoSystems - 49500 рублей
- Курс для разработчика Stepik - Бесплатно
- Youtube канал `aurora os community`
- Youtube канал KDAВ - эксперты в разработке Qt

The screenshot shows the YouTube channel page for KDAВ, which has 8.5 thousand subscribers. The page features a grid of 18 video thumbnails. Each thumbnail includes a video player preview, a title, and view/like statistics. The videos are related to Qt development, Qt Widgets, and Qt Quick. The channel name 'KDAВ' is prominently displayed at the top, along with a 'ПОДПИСАТЬСЯ' (Subscribe) button. Navigation tabs for 'ГЛАВНАЯ', 'ВИДЕО', 'ПЛЕЙЛИСТЫ', 'СООБЩЕСТВО', 'КАНАЛЫ', and 'О КАНАЛЕ' are visible. A 'Загрузки' (Downloads) section is also present at the top right of the video grid.

Video Title	Views	Time
QtDevCon22 – QML Hot Reload in Practice: How to...	206 просмотров	3 дня назад
Save Re-compile Time - Include moc Files in Source...	309 просмотров	4 дня назад
QtDevCon22 – Closing the Gaps – QML on Desktop	376 просмотров	5 дней назад
QtDevCon22 – How Can I Make My Qt Apps More...	578 просмотров	10 дней назад
QStringBuilder	351 просмотр	11 дней назад
QtDevCon22 – 25k Stars Later – How to Survive...	196 просмотров	12 дней назад
Handling Qt based embedded devices up to date	53-05	
Qt Widgets and more	28-26	
KDE's journey to Qt 6	51-53	
Five for me! Goodies from QtDevCon22	9-48	
Handling Qt-based embedded devices up to date	9-47	
Qt Widgets and more	29-04	
QtDevCon22 – Keeping Qt Based Embedded Devices u...	207 просмотров	2 недели назад
Which String Class in Qt Should I Use?	429 просмотров	2 недели назад
QtDevCon22 – KDE's Journey to Qt6	409 просмотров	2 недели назад
QtDevCon22 – Five (or More) Goodies from...	242 просмотра	3 недели назад
QtDevCon22 – Handling Large Amount of Text in QML	349 просмотров	3 недели назад
Editing Delegates	262 просмотра	3 недели назад
QtDevCon22 – Set up a Shareable Dev Environment...	218 просмотров	3 недели назад
QtDevCon22 – Why Energy Consumption of Digitization...	83 просмотра	3 недели назад
QtDevCon22 – Community Driven QML Coding...	668 просмотров	1 месяц назад
QtDevCon22 – Qt them Delegates - Some Real World Examples	344 просмотра	1 месяц назад
QtDevCon22 – Proteins, Microscopes and Qt – How...	196 просмотров	1 месяц назад
KDAВ Interview	525 просмотров	1 месяц назад



Setup проекта



Разработка

Qt / QML

Qt

- Фреймворк на языке C++

Qt

- Фреймворк на языке C++
- Полностью кроссплатформенный (поддерживает Android, iOS, Desktop)

Qt

- Фреймворк на языке C++
- Полностью кроссплатформенный (поддерживает Android, iOS, Desktop)
- Огромное коммьюнити

Qt

- Фреймворк на языке C++
- Полностью кроссплатформенный (поддерживает Android, iOS, Desktop)
- Огромное коммьюнити
- Большое количество готовых решений

QML (Qt Meta Language)

- Декларативный язык программирования

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

- Декларативный язык программирования
- Содержит элементы JavaScript

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

- Декларативный язык программирования
- Содержит элементы JavaScript
- Совместим с C++

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```

QML (Qt Meta Language)

```
Page {  
    objectName: "aboutPage"  
    allowedOrientations: Orientation.All  
  
    Label {  
        objectName: "descriptionText"  
        anchors {  
            left: parent.left;  
            right: parent.right;  
            margins: Theme.horizontalPageMargin  
        }  
        color: palette.highlightColor  
        font.pixelSize: Theme.fontSizeSmall  
        textFormat: Text.RichText  
        wrapMode: Text.WordWrap  
        text: qsTr("#descriptionText")  
    }  
}
```


QML + JavaScript

```
Row {  
    id: labelsRow  
  
    property alias count: labelsRepeater.count  
    property alias model: labelsRepeater.model  
  
    function modelGet(index) {  
        return labelsRepeater.model.get(index)  
    }  
  
    Repeater {  
        id: labelsRepeater  
        anchors.verticalCenter: parent.verticalCenter  
  
        onClicked: { labelsRow.selectLabel(index) }  
    }  
}
```

QML + JavaScript

```
Row {  
    id: labelsRow  
  
    property alias count: labelsRepeater.count  
    property alias model: labelsRepeater.model  
  
    function modelGet(index) {  
        return labelsRepeater.model.get(index)  
    }  
  
    Repeater {  
        id: labelsRepeater  
        anchors.verticalCenter: parent.verticalCenter  
  
        onClicked: { labelsRow.selectLabel(index) }  
    }  
}
```

QML + JavaScript

```
Row {  
    id: labelsRow  
  
    property alias count: labelsRepeater.count  
    property alias model: labelsRepeater.model  
  
    function modelGet(index) {  
        return labelsRepeater.model.get(index)  
    }  
  
    Repeater {  
        id: labelsRepeater  
        anchors.verticalCenter: parent.verticalCenter  
  
        onClicked: { labelsRow.selectLabel(index) }  
    }  
}
```

QML + JavaScript

```
Row {  
    id: labelsRow  
  
    property alias count: labelsRepeater.count  
    property alias model: labelsRepeater.model  
  
    function modelGet(index) {  
        return labelsRepeater.model.get(index)  
    }  
  
    Repeater {  
        id: labelsRepeater  
        anchors.verticalCenter: parent.verticalCenter  
  
        onClicked: { labelsRow.selectLabel(index) }  
    }  
}
```

Навигация

```
onClicked: {  
    pageStack.push(Qt.resolvedUrl("OptionsPage.qml"))  
}
```

```
onClicked: {  
    pageStack.replace(Qt.resolvedUrl("OptionsPage.qml"))  
}
```

```
onClicked: {  
    pageStack.pop()  
}
```

Бизнес-логика

Бизнес-логика



- Jetpack Compose
- ViewModel
- LiveData
- Coroutines
- Clean Arch

Бизнес-логика



- Jetpack Compose
- ViewModel
- LiveData
- Coroutines
- Clean Arch



- SwiftUI
- ViewModel
- Combine
- Async/Await
- Clean Arch



Бизнес-логика



- Jetpack Compose
- ViewModel
- LiveData
- Coroutines
- Clean Arch



- SwiftUI
- ViewModel
- Combine
- Async/Await
- Clean Arch



- QML
- ???

Бизнес-логика. JavaScript



```
Component.onCompleted: {  
  var http = new XMLHttpRequest()  
  var url = "https://google.com"  
  http.open("GET", url, true)  
  
  http.setRequestHeader("Content-type",  
    "application/x-www-form-urlencoded");  
  http.onreadystatechange = function() {  
    if (http.readyState == 4) {  
      if (http.status == 200) {  
        console.log("ok")  
      } else {  
        console.log("error: " + http.status)  
      }  
    }  
  }  
}
```

Бизнес-логика. C++



- Jetpack Compose
- ViewModel
- LiveData
- Coroutines
- Clean Arch



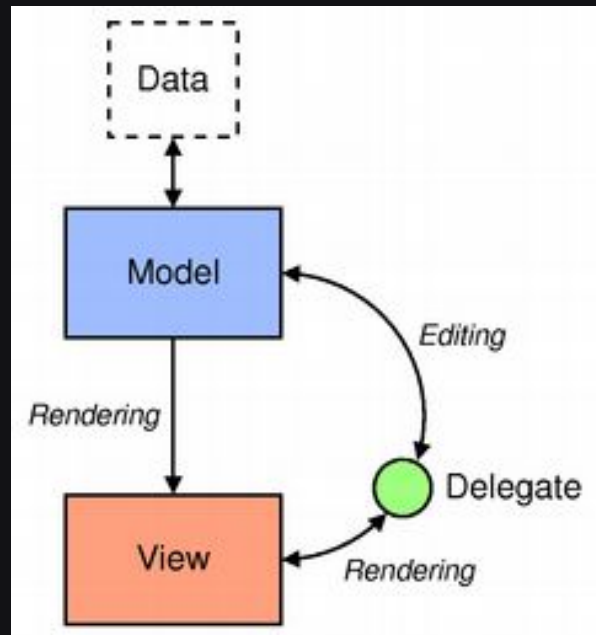
- SwiftUI
- ViewModel
- Combine
- Async/Await
- Clean Arch



- QML
- MVC, Delegate
- Signals/Slots
- Qt Concurrent, QFuture

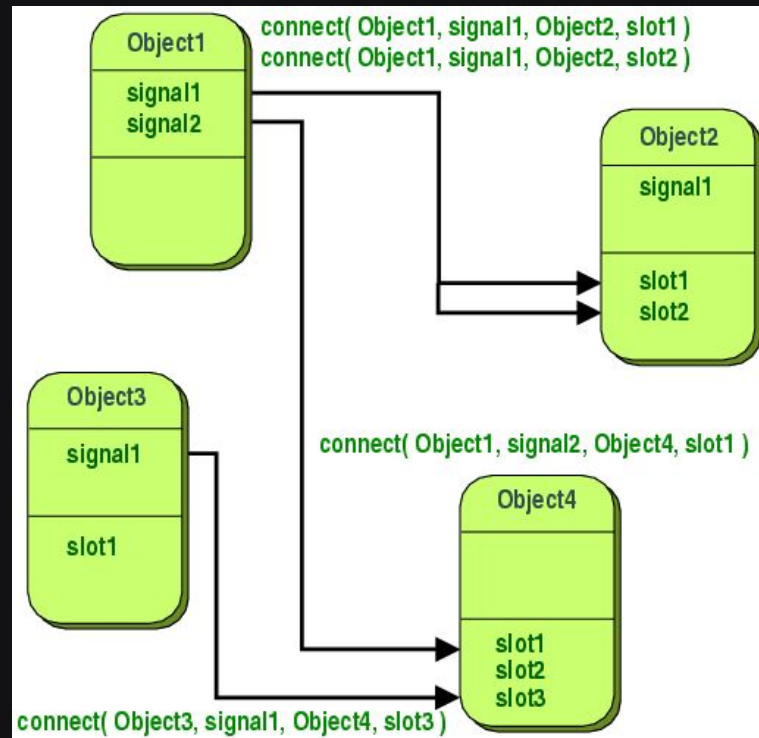
Бизнес-логика. MVC

- Не тот MVC, что в Android/iOS
- Model похожа на презентер
- Существует система Observable/Observer



Бизнес-логика. Signals/Slots

- Концепция MVVM
- Изменения состояния ведут к сигналу
- Слот ловит изменения сигнала



Бизнес-логика. Signals/Slots

```
class Counter : public QObject {
    Q_OBJECT

public:
    Counter() { m_value = 0; }

    int value() const { return m_value; }

public slots:
    void setValue(int value);

signals:
    void valueChanged(int newValue);

private:
    int m_value;
};
```

Бизнес-логика. Signals/Slots

```
class Counter : public QObject {
    Q_OBJECT

public:
    Counter() { m_value = 0; }

    int value() const { return m_value; }

public slots:
    void setValue(int value);

signals:
    void valueChanged(int newValue);

private:
    int m_value;
};
```

Бизнес-логика. Signals/Slots

```
class Counter : public QObject {
    Q_OBJECT

public:
    Counter() { m_value = 0; }

    int value() const { return m_value; }

public slots:
    void setValue(int value);

signals:
    void valueChanged(int newValue);

private:
    int m_value;
};
```


Бизнес-логика. Signals/Slots

```
class Counter : public QObject {
    Q_OBJECT

public:
    Counter() { m_value = 0; }

    int value() const { return m_value; }

public slots:
    void setValue(int value);

signals:
    void valueChanged(int newValue);

private:
    int m_value;
};
```

Бизнес-логика. Signals/Slots

```
// main.cpp
```

```
Counter counter;  
context->setContextProperty("counter", &counter);
```

Бизнес-логика. Signals/Slots

```
// main.cpp
```

```
Counter counter;  
context->setContextProperty("counter", &counter);
```

```
// SomeScreen.qml  
// Connect QML to C++  
Connections {  
    target: counter  
}
```

Бизнес-логика. Signals/Slots

```
// main.cpp

Counter counter;
context->setContextProperty("counter", &counter);

// SomeScreen.qml
// Connect QML to C++
Connections {
    target: counter
}

// Use C++ class from QML
Button {
    text: qStr("Set counter value")
    onClicked: counter.setValue(20) // Slot call
}
```

Бизнес-логика. C++

- QFuture система похожа на Rx

Бизнес-логика. C++

- QFuture система похожа на Rx
- Есть возможность работать с БД через API Qt

Бизнес-логика. C++

- QFuture система похожа на Rx
- Есть возможность работать с БД через API Qt
- Работа с изображениями (как с интернета, так и локальными)

Бизнес-логика. C++

- QFuture система похожа на Rx
- Есть возможность работать с БД через API Qt
- Работа с изображениями (как с интернета, так и локальными)
- Легкая работа с анимациями

Минусы

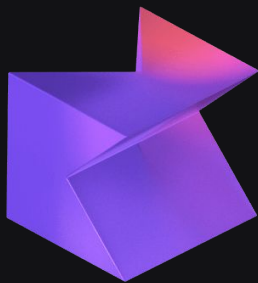
Минусы

- С++ сложный



Минусы

- C++ сложный



+



+



Минусы

- C++ сложный
- Немного придется изменить ментальную модель



Минусы

- C++ сложный
- Немного придется изменить ментальную модель
- Нет легкой возможности получить устройство для теста

Aurora Developer's Beta

Aurora Dev Beta

> Мобильное устройство на базе Aurora OS

Aurora Dev Beta

- > Мобильное устройство на базе Aurora OS
- > Доступ в Aurora Market

Aurora Dev Beta

- > Мобильное устройство на базе Aurora OS
- > Доступ в Aurora Market
- > Чат с поддержкой

Aurora Dev Beta

- > Мобильное устройство на базе Aurora OS
- > Доступ в Aurora Market
- > Чат с поддержкой
- > Community разработчиков

Всего лишь 10 человек!

НЕ НУ

А ЧЁ ТАК МАЛО?

Как попасть

> Написать на beta@omp.ru

Как попасть

- > Написать на beta@omp.ru
- > Получить анкету и заполнить ее

Как попасть


- > Написать на beta@omp.ru
- > Получить анкету и заполнить ее
- > Дождаться приглашения

Как попасть

- > Написать на beta@omp.ru
- > Получить анкету и заполнить ее
- > Дождаться приглашения
- > Подписать документы, получить все девайсы и вступить в команду


UX

Результат

TESLA

1. Car2. Exterior3. Interior4. Autopilot

Select your car



PerformanceLong Range


\$55,700\$46,700

3.5s150mph

0-60 mphTop Speed


Tesla All-Wheel Drive has two independent motors. Unlike traditional all-wheel drive systems, these two motors digitally control torque to the front and rear wheels—for far better handling and traction control.

\$55,700NEXT

TESLA


1. Car2. Exterior3. Interior4. Autopilot

Select color




Sad Blue

\$1,800




20" Performance WheelsCarbon fiber spoiler

\$57,500NEXT

TESLA


1. Car2. Exterior3. Interior4. Autopilot




Select interior

Black and WhiteAll Black Included


\$1,000



\$58,500NEXT

TESLA

1. Car2. Exterior3. Interior4. Autopilot



Autopilot

Full Self-DrivingAutopilot

\$5,000\$3,000

Automatic driving from highway on-ramp to off-ramp including interchanges and overtaking slower cars.

\$63,500NEXT

Спасибо за внимание

Вопросы?

Контакты для связи

Email: bob298@yandex.ru

Youtube: MobileDeveloper

Telegram: [@mobiledevnews](https://t.me/mobiledevnews)