

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ НЕФТИ И ГАЗА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)
ИМЕНИ И.М. ГУБКИНА»

ФАКУЛЬТЕТ КОМПЛЕКСНОЙ БЕЗОПАСНОСТИ ТЭК
КАФЕДРА БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Лабораторная работа №16
по дисциплине «Специализированные языки и технологии
программирования»
на тему «Отладка и профилирование с помощью Aurora SDK»

Выполнил студент:
группы КА-22-06
Воронин Алексей Дмитриевич

Преподаватель:
Греков Владимир Сергеевич

Москва, 2026

Оглавление	
Цель работы	3
Задание 1: Подготовка проекта.....	4
Задание 2: Работа с режимами выполнения	4
Задание 3: Отладка приложения.....	4
Задание 4: Модификация параметров	5
Задание 5: Рефакторинг кода	6
ЗАКЛЮЧЕНИЕ	8
Контрольные вопросы	9

Цель работы

1. Освоить базовые методы отладки QtQuick-приложений
2. Научиться работать с точками останова
3. Получить навыки модификации параметров приложения во время выполнения
4. Изучить принципы рефакторинга исходного кода

Задание 1: Подготовка проекта

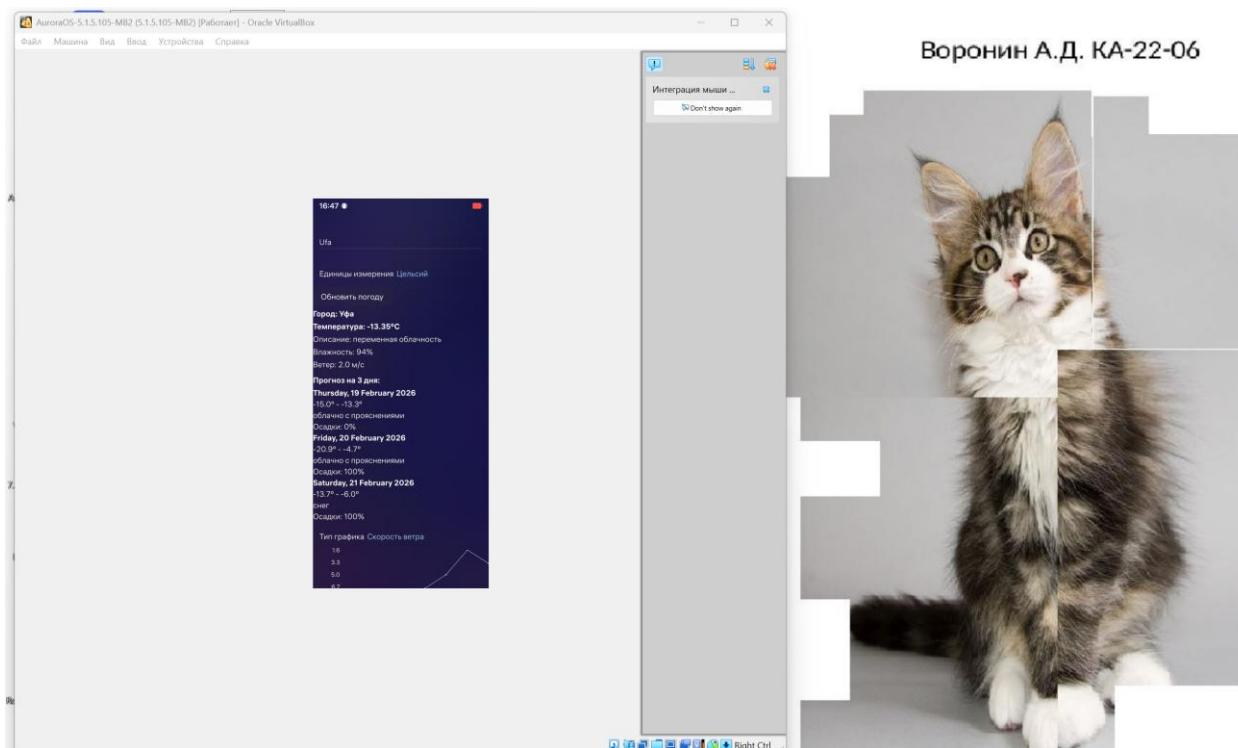
- Создать новое QtQuick-приложение на основе предыдущих лабораторных работ (2-10)
- Проверить работоспособность базового функционала

В качестве исходного проекта была выбрана лабораторная работа №6 «Разработка приложения на QML». В данной лабораторной работе нужно было разработать приложение "Погодный информатор".

Задание 2: Работа с режимами выполнения

- Запустить приложение в обычном режиме выполнения
- Проверить реакцию на пользовательский ввод
- Зафиксировать базовые параметры (размер окна, заголовок)

Приложение запускается. Пользователь вводит желаемый город и видит прогноз погоды после нажатия на кнопку «Обновить погоду». Также при выборе единицы измерения температуры и типа графика окно обновляется.

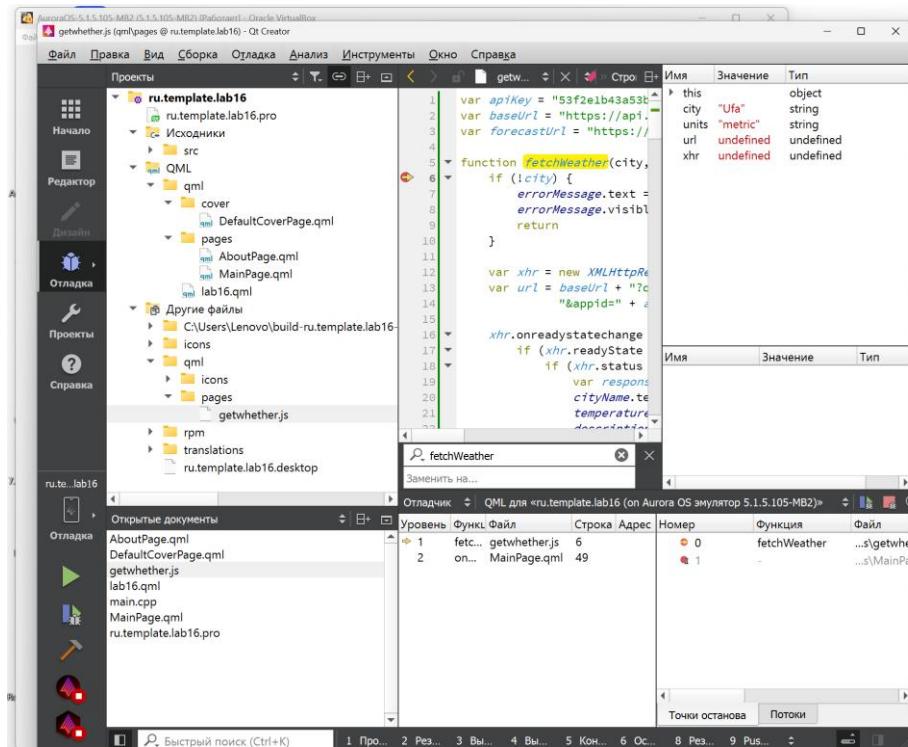


Задание 3: Отладка приложения

1. Установить точки останова в ключевых местах кода
2. Запустить приложение в режиме отладки

3. Выполнить пошаговую отладку основных функций
4. Просмотреть значения переменных в точках останова

Установим точку останова в функции `fetchWeather` и пронаблюдаем значения.



The screenshot shows the Qt Creator IDE interface. The central part is the code editor with the file `getwhether.js` open. A red dot indicates a break point is set on the first line of the `fetchWeather` function. To the right of the code editor is a variable watch window titled "Имя" (Name) showing local variables: `this` (object), `city` ("Ufa") (string), `units` ("metric") (string), `url` (undefined), and `xhr` (undefined). Below the code editor is a search bar containing "fetchWeather". At the bottom of the screen, there is a status bar with various icons and text.

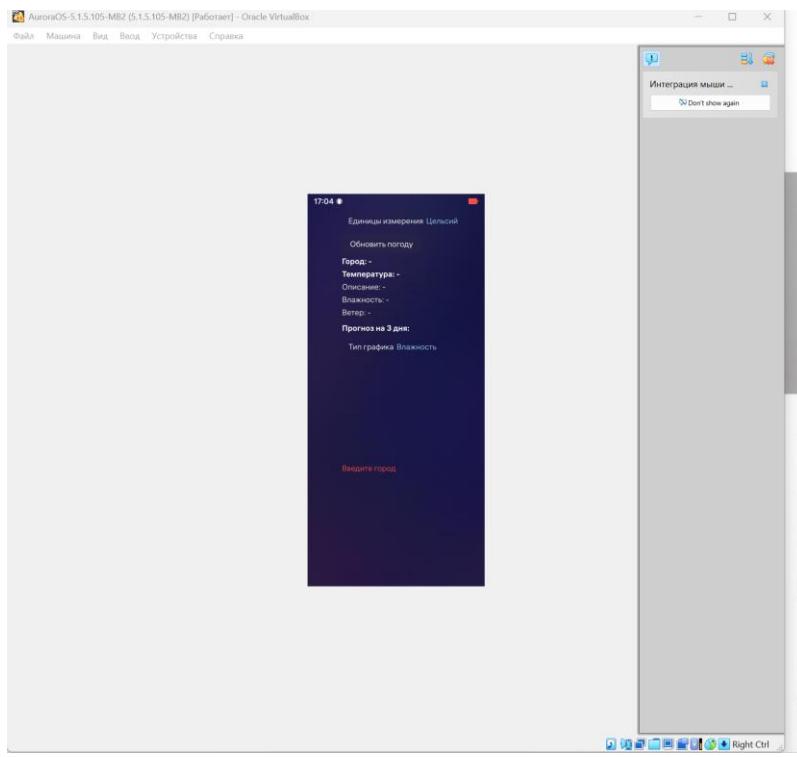
Воронин А.Д. КА-22-06



Задание 4: Модификация параметров

- Изменить заголовок окна приложения через свойства QML
- Откорректировать размеры главного окна
- Проверить изменения в режиме выполнения

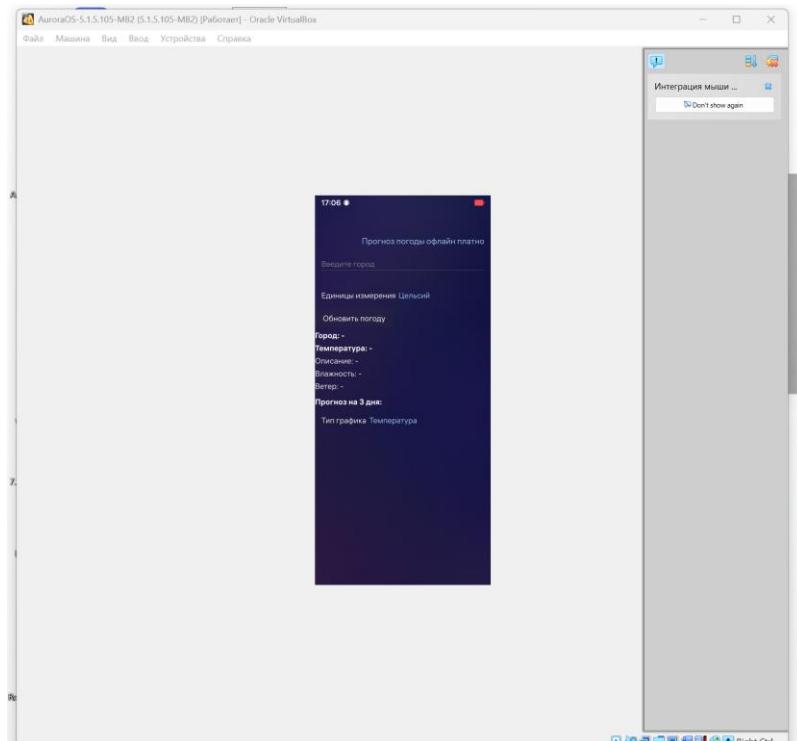
При попытке изменить размер приложения оно ломается. Пользователю становится недоступна часть интерфейса.



Воронин А.Д. КА-22-06



Изменим заголовок.



Воронин А.Д. КА-22-06



Задание 5: Рефакторинг кода

- Переименовать основные .cpp и .qml файлы
- Обновить ссылки в проекте
- Проверить работоспособность после изменений

Заменим MainPage на MainPageQML.

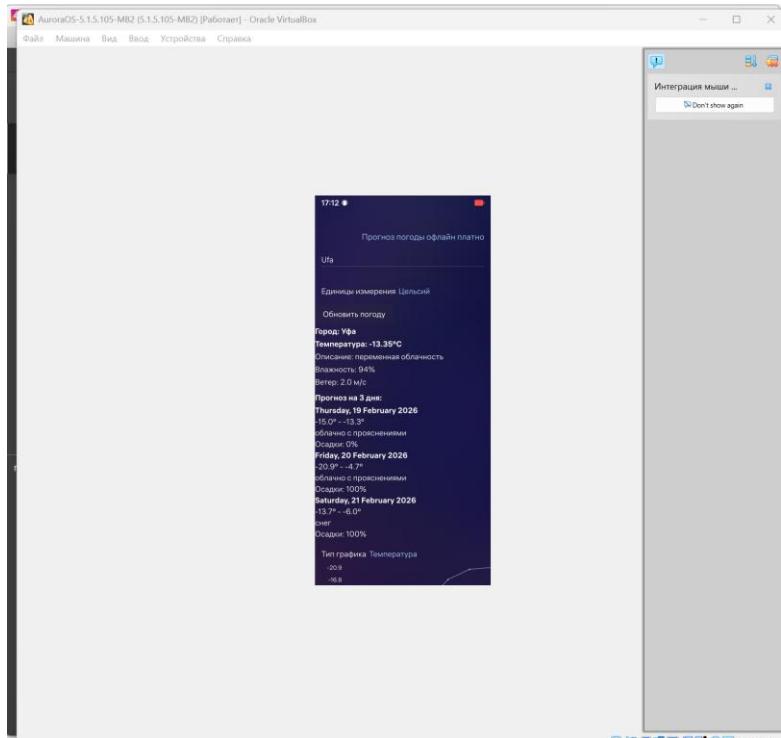
The screenshot shows the Qt Creator IDE interface. The top menu bar includes 'Файл', 'Правка', 'Вид', 'Сборка', 'Отладка', 'Анализ', 'Инструменты', 'Окно', and 'Справка'. The title bar indicates the project is 'lab16.qml (qml @ ru.template.lab16) - Qt Creator'. The left sidebar has icons for 'Начало', 'Редактор', 'Дизайн', 'Отладка', 'Проекты', and 'Справка'. The main area shows a file tree for 'ru.template.lab16' with 'ru.templ' expanded, showing 'Исходный код' (src), 'QML' (expanded), and 'qm' (expanded). Below the file tree is a preview pane showing a QML component named ' MainPage'. The bottom part of the interface features a terminal window with the following log output:

```
https://developer.auroraos.ru/doc/software_development/guides/package_signing
17:06:15: Развертывание RPM-пакета...
17:06:15: Запускается: «C:\AuroraOS\bin\sfdk.exe» -c "device=Aurora OS
эмплатор 5.1.5.105-MB2" deploy --sdk --silent --keepUserData C:
\Users\Lenovo\build-ru.template.lab16-
AuroraOS_5_1_5_105_MB2_x86_64_in_Aurora_Build_Engine_5_1_5_105_mb2-
Debug\RPMS\ru.template.lab16-0.1-1.x86_64.rpm
sfdk: [I] PHPTCIPERPC PEP: C:\CSCP\RPC: -c target=AuroraOS-5.1.5.105-MB2-x86_64
Installing ru.template.lab16-0.1-1.x86_64.rpm
Installation successful
17:06:21: Процесс «C:\AuroraOS\bin\sfdk.exe» завершился успешно.
17:06:21: Прошло времени: 00:27.
```

Воронин А.Д. КА-22-06



Проверим работоспособность.



Воронин А.Д. КА-22-06



ЗАКЛЮЧЕНИЕ

В заключение, подводя итог проделанной работе, все поставленные в начале цели и задания данной лабораторной работы были достигнуты в полном объеме.

Контрольные вопросы

1. Какие режимы выполнения приложений существуют в Qt Creator?

В Qt Creator доступны следующие основные режимы выполнения:

Запуск – компиляция и запуск приложения без привязки отладчика.
Используется для быстрой проверки.

Отладка – запуск с подключённым отладчиком, позволяющим устанавливать точки останова, выполнять пошаговое исполнение и просматривать переменные.

Профилирование (Profile) – сбор данных о производительности (использование CPU, памяти и т.д.) для оптимизации.

2. Как установить точку останова в QML-коде?

Точку останова в QML можно установить, кликнув слева от нужной строки в редакторе - появится красная точка.

3. Какие способы изменения свойств окна вы использовали?

Прямое задание размеров:

width: 1000

height: 1000

4. Какие проблемы могут возникнуть при переименовании файлов?

Нарушение путей в импортах – если файл подключается через import или include, необходимо обновить эти ссылки во всех файлах, где он используется.

Если файл содержит относительные пути к другим файлам, они могут перестать работать.