

Курсовой проект

Редактор Sequence диаграмм

Sequence Diagram Editor

Главный разработчик: Казаков А.В.

Разработчик: Проводин Р.А.

Менеджер проекта и документооборота: Цвеков И.И.



Цель



Упростить создания диаграмм
последовательности. По средствам
создания инструмента -
графического редактора диаграмм
последовательности.

Задачи

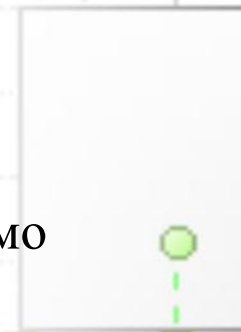
- Изучение Sequence диаграмм
- Анализ и выбор методов для создания программного продукта
- Создание программного продукта
- Внедрение программного продукта



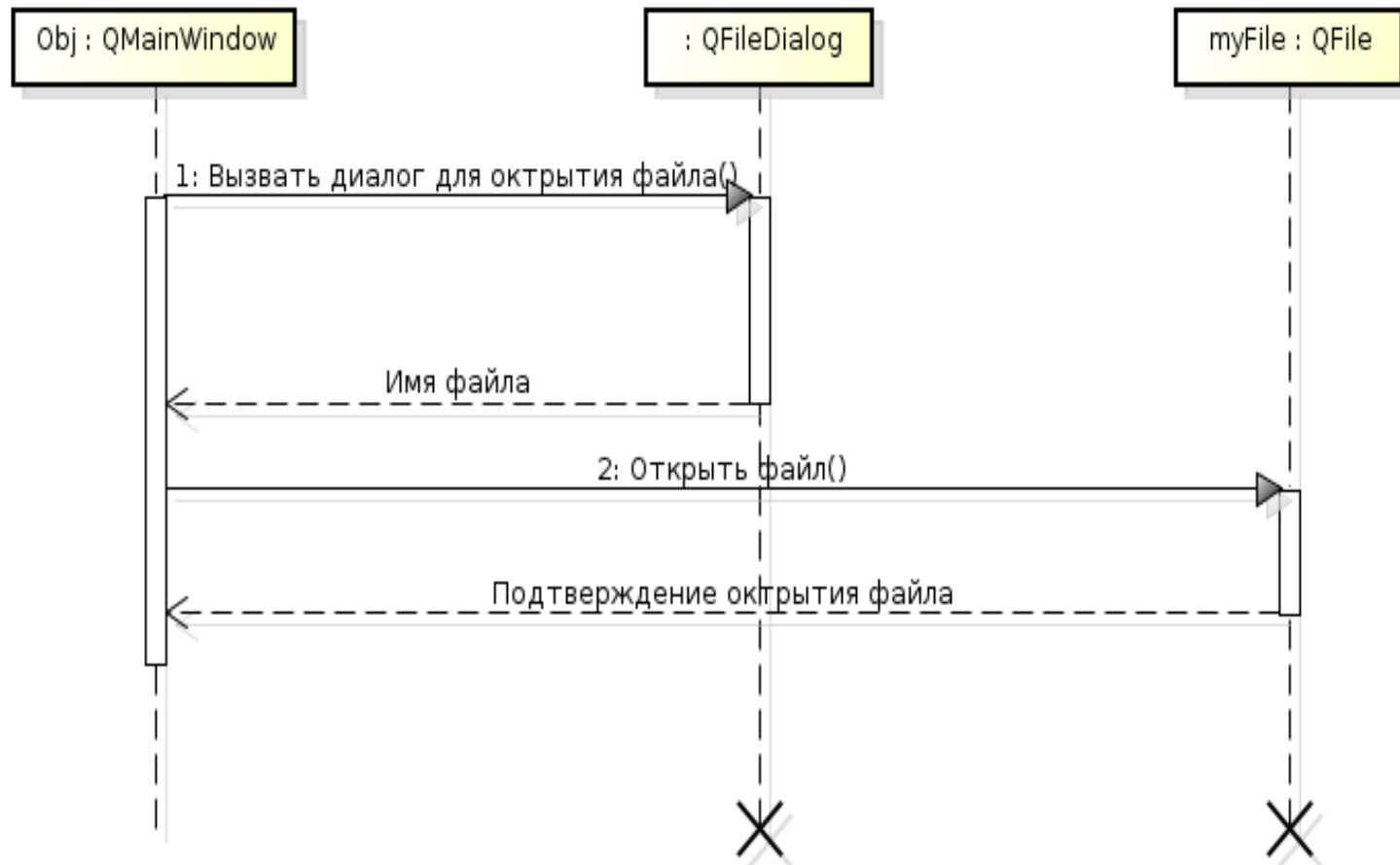
Sequence диаграмма

Диаграммы взаимодействия (interaction diagrams) описывают взаимодействие групп объектов в различных условиях их поведения. UML определяет диаграммы взаимодействия нескольких типов, из которых наиболее употребительными являются диаграммы последовательности. Обычно диаграмма последовательности описывает один сценарий. На диаграмме показаны экземпляры объектов и сообщения, которыми обмениваются объекты в рамках одного прецедента (use case).

Основными элементами диаграммы последовательности являются обозначения объектов (прямоугольники), вертикальные линии (англ. lifeline), отображающие течение времени при деятельности объекта, и стрелки, показывающие выполнение действий объектами. На данной диаграмме объекты располагаются слева направо. Ее недостатком является то, что она занимает много места.



Пример диаграммы



Принятые решения

1. На уровне проектирования функциональности программного продукта

- Прецеденты можно посмотреть в Use-Case диаграмме
- Структуру классов можно посмотреть на диаграмме классов
- Ориентация на разработчиков малых программных систем

2. На уровне организации процесса разработки

- Использование технологий, с которыми знакома вся команда(Qt)
- Не разграничивание участников по ролям в процессе разработки(каждый участвовал во всех стадиях проекта)
- Последовательное создание документации в процессе разработки
- Использование средств связи: Skype
- Использование средства командной разработки: mercurial
- Использование средства синхронизации процесса разработки: bitbucket



Use-Case диаграмма

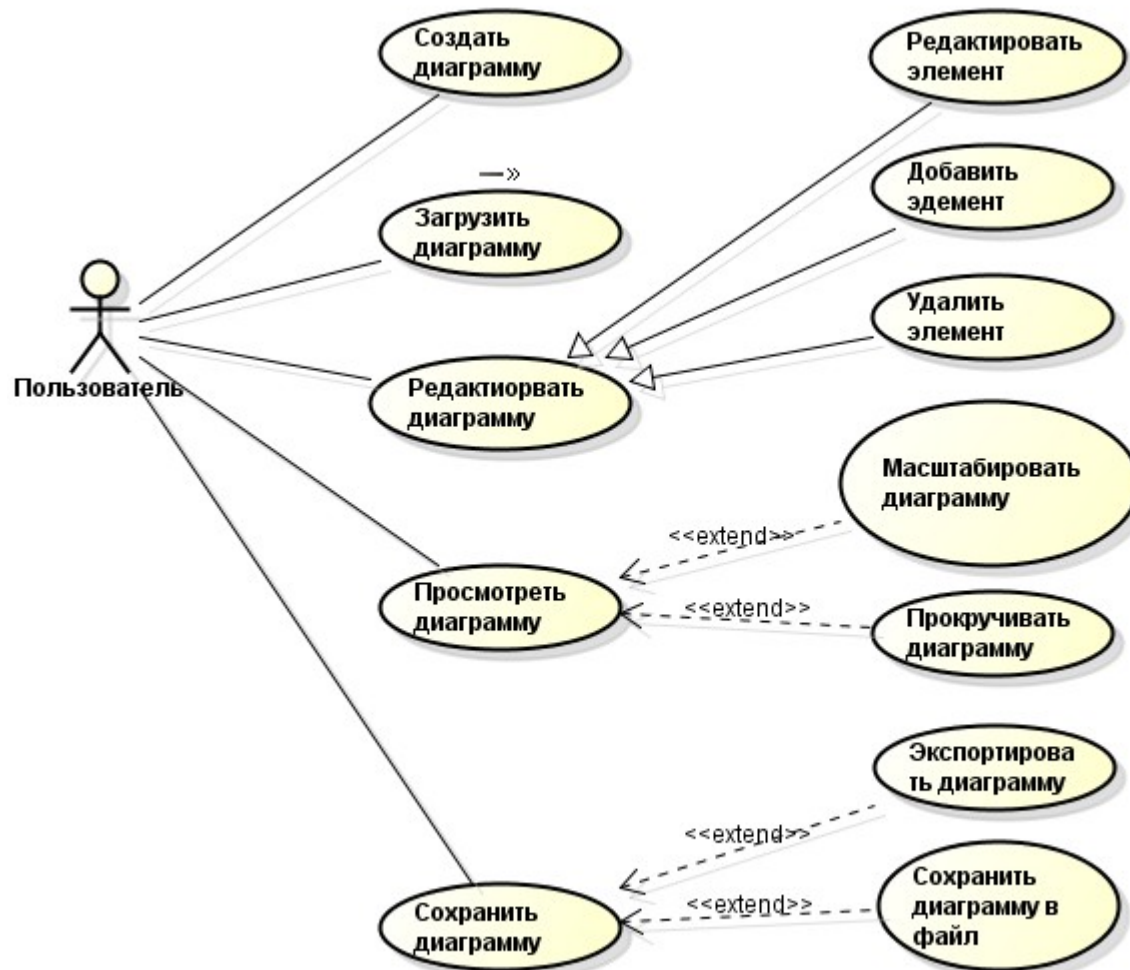
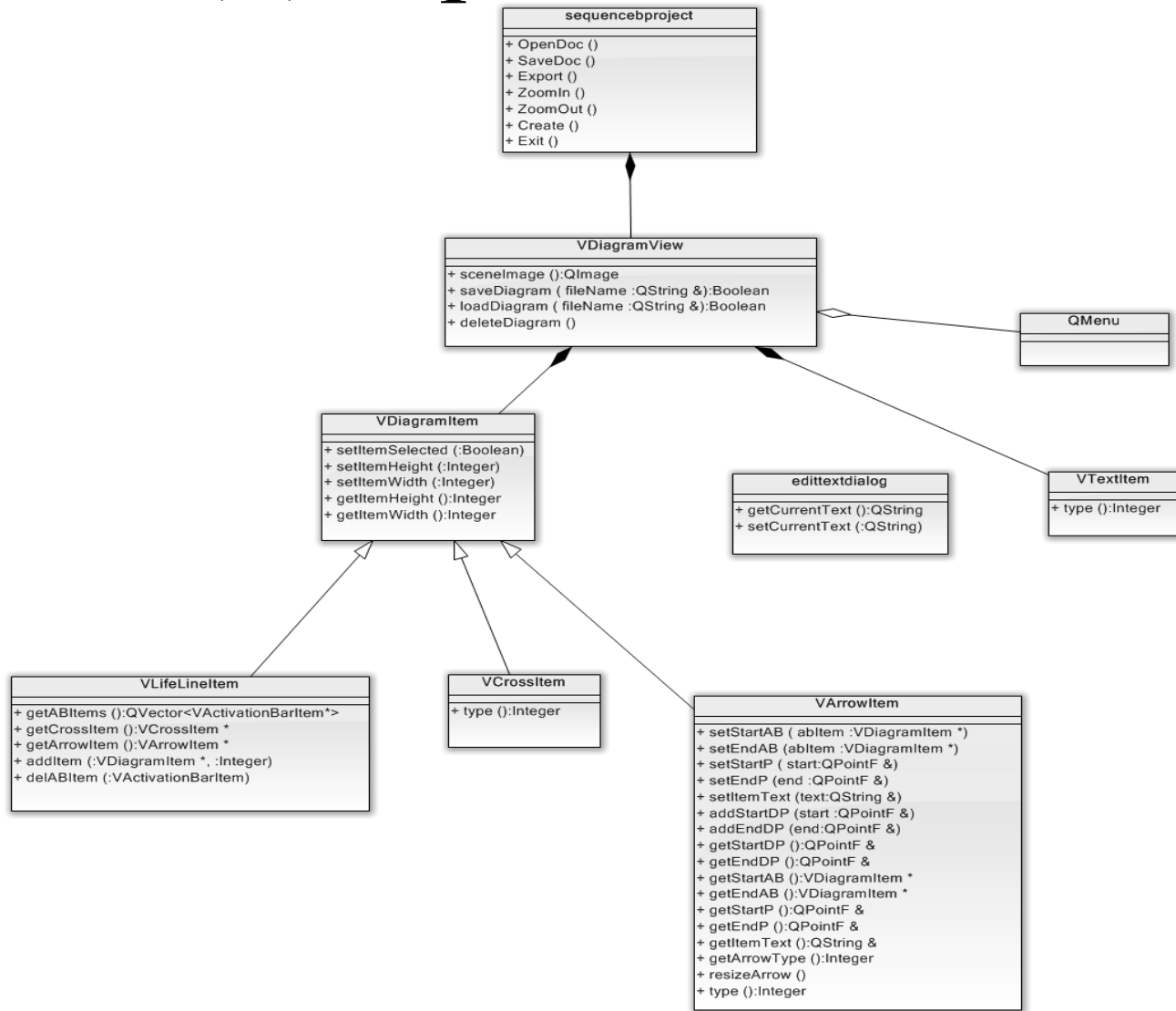


Диаграмма классов



Сравнение диаграмм разных программ

Диаграмма построенная в стороннем редакторе

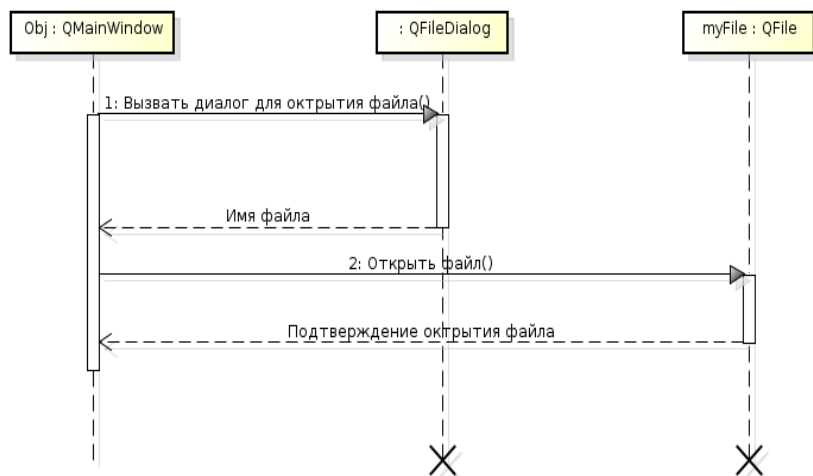
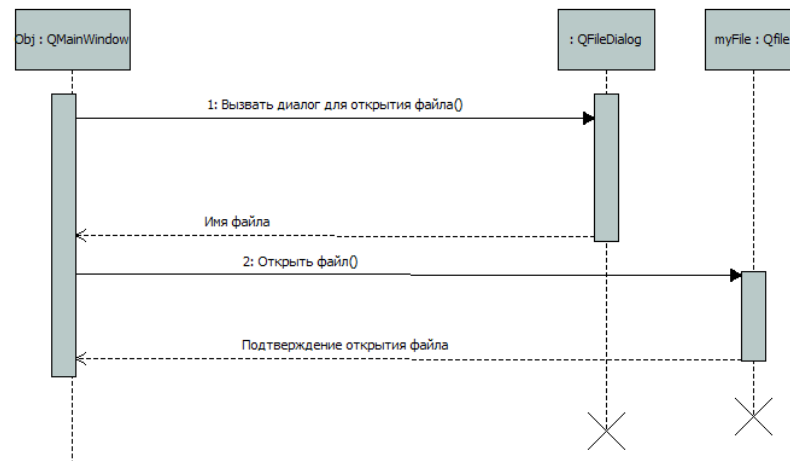


Диаграмма построенная SDE



Личный вклад

#	Казаков А.В.	Проводин Р.А.	Цветков И.И.
Разработка требований	Описание требований	Создание макета программы	Создание интерфейса программы
Разработка проектных решений	Разработка диаграммы классов	Разработка диаграммы классов	Разработка Use-case диаграммы и диаграммы Activity
Кодировка программы	Кодирование программы	Кодирование программы	Кодирование программы
Тестирование	Разработка методики тестирования. Написание тестов.	Написание тестов.	Написание тестов.
Внедрение программы	Написание руководство пользователя и системного программиста	Написание инсталлятора	Написание презентации

Результаты

1. Надёжность.

Программа не надёжна и проходит не все существующие тесты

2. Функциональность

Программа реализует весь функционал, заявленный в постановке задачи

3. Эффективность

Продукт позволяет эффективно создавать диаграммы



Отличие от аналогов

Простота использования - достигается ,главным образом ,за счёт простого , интуитивно понятного интерфейса

Узкоспециализированность(создание только sequence — диаграмм) – если человеку необходимо создать именно диаграмму последовательности то гораздо выгодней использовать специализированный редактор , нежели разбираться в устройстве сложного универсального редактора

Дальнейшие планы

- Реализация истории изменений
- Множественное выделение элементов, дублирование элементов при перетаскивании с зажатой клавишей Ctrl
- Копирование, вырезание и вставка выделенных элементов, включая копирование-вставку между несколькими экземплярами запущенной программы



Спасибо за внимание

