

Санкт–Петербургский государственный университет
Математическое обеспечение и администрирование информационных систем

Ким Юния Александровна

Веб-редактор для REAL.NET

Отчёт по учебной практике

Научный руководитель:
к. т. н., доцент Литвинов Ю. В.

Санкт-Петербург
2019 г.

Содержание

| | |
|---|----|
| Введение | 3 |
| Постановка задачи | 4 |
| 1. Обзор существующих решений | 5 |
| 1.1. Web Modeling Project | 5 |
| 1.2. Draw.io | 8 |
| 1.3. Архитектура редактора | 8 |
| 2. Описание решения | 10 |
| Заключение | 13 |
| Список литературы | 14 |

Введение

В современном мире потребность в упрощении использования различных технологий растет с каждым днем. На данный момент существует несколько инструментов, которые могут решить данную проблему. Такими инструментами можно считать и платформы для работы с предметно-ориентированными визуальными языками программирования. Предметно-ориентированный язык программирования – язык, используемый для решения задач в конкретной области. Визуальный язык программирования – язык, позволяющий создавать программы с помощью манипуляций графическими элементами без печати текстовых команд. Одной из таких платформ является REAL.NET [1] – среда для создания визуальных предметно-ориентированных языков программирования.

Однако на данный момент не существует никаких веб-редакторов, которые позволили бы работать с языками, созданными с помощью REAL.NET. В отличие десктопных программ, их веб-аналоги не нужно скачивать, что является огромным плюсом для пользователей. Именно поэтому появилась потребность в создании подобного редактора.

Данный проект является групповым. Автору отчета предлагалось внедрить поддержку операций отмены и возврата в веб-редактор, а также реализовать палитру элементов редактора. Палитра – часть веб-редактора, на которой отображаются все элементы, перетаскиванием которых можно создавать модели на сцене.

Постановка задачи

Целью данной работы является внедрение поддержки операций отмены и возврата (Undo/Redo) в веб-редактор, а также реализация палитры элементов редактора.

Для достижения данных целей были поставлены следующие задачи.

- Изучение языка программирования TypeScript.
- Обзор существующих веб-редакторов.
- Внедрение поддержки операций отмены и возврата в веб-редактор.
- Реализация палитры элементов редактора.

1. Обзор существующих решений

В данном разделе приводится обзор некоторых существующих веб-редакторов, а также анализ их реализаций операций отмены и возврата и палитры элементов редактора.

1.1 Web Modeling Project

Web Modeling Project (WMP) [2] – среда разработки решений различных задач с помощью предметно-ориентированных визуальных языков программирования. Его редактор был написан на языке TypeScript с использованием паттерна Model-View-Controller. Данный шаблон разделяет веб-приложение на три компонента: модель (model), представление (view) и контроллер (controller). Благодаря этому работа и модификация каждой составляющей может осуществляться независимо.

Модель представляет собой компонент, отвечающий за хранение и обработку данных системы. Она не взаимодействует с пользователем напрямую, а лишь реагирует на запросы контроллера, передавая ему в ответ определенные данные. Представление отвечает за отображение информации, то есть внешний вид. Контроллер в свою очередь соединяет модель и представление, а также обеспечивает связь между программой и пользователем, обрабатывая запросы последнего. Графическая интерпретация работы паттерна Model-View-Controller представлена на рис. 1.

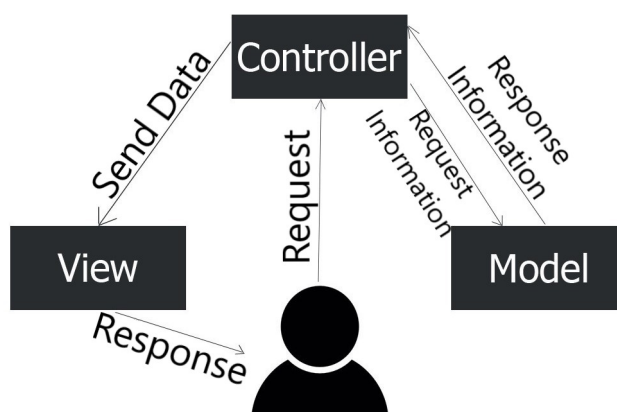


Рис. 1: Паттерн Model-View-Controller (из [6])

Палитра веб-редактора WMP также была разделена на три вышепере-

численных компонента. Модель палитры хранит различные типы элементов, которыми можно манипулировать. Представлением в данном случае являются HTML-шаблоны, которые отображаются на веб-странице в виде палитры и ее элементов. Что касается контроллера, то он оперирует поведением и реакцией элементов на различные действия со стороны пользователя.

Говоря об операциях отмены и возврата в редакторе WMP, нужно упомянуть еще один общеизвестный паттерн, который называется Command. Суть паттерна заключается в инкапсулировании запросов в отдельные объекты, которые называют командами. Для этого обычно используется интерфейс, реализуемый каждой конкретной операцией. Таким образом, чтобы внедрить в систему новую операцию, достаточно реализовать данный интерфейс. Графическая интерпретация паттерна Command представлена на рис. 2.

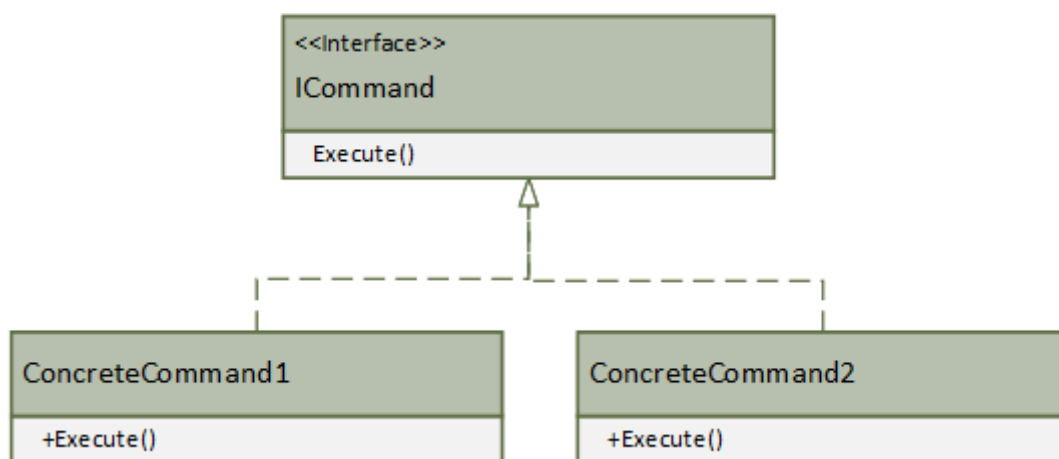


Рис. 2: Паттерн Command (из [7])

Для реализации операций отмены и возврата в редакторе WMP был реализован интерфейс Command, который реализует каждая конкретная команда в редакторе, например, MoveElementCommand, CreateElementCommand, ChangeCurrentElementCommand и многие другие. Помимо этого имеется также класс-контроллер UndoRedoController, который отвечает за вызов операций отмены и возврата, а также хранение выполняемых пользователем команд. У класса имеется список, в котором он хранит все совершаемые команды, указатель на одну из ячеек списка, его максимальный размер, а также несколько методов, которые добавляют, отменяют или возвращают команду и

метод для удаления нескольких элементов списка. В зависимости от того, какие команды совершаются в процессе работы, указатель списка перемещается либо в сторону начала, либо в сторону конца списка. При вызове метода undo или redo совершается отмена или возврат той команды, на которую показывает указатель. Графическая интерпретация реализации undo/redo в редакторе WMP представлена на рис. 3. Пример работы класса UndoRedoController в WMP можно увидеть на рис. 4.

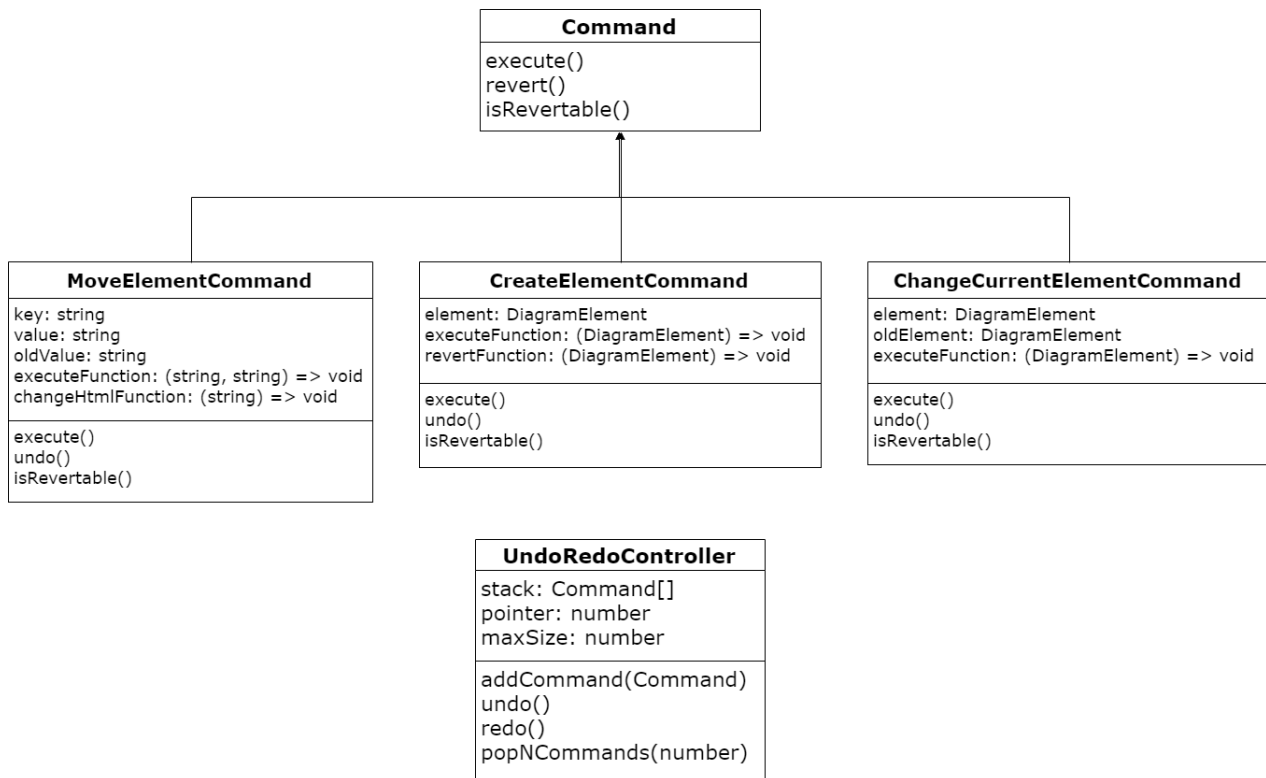


Рис. 3: Undo/Redo в WMP

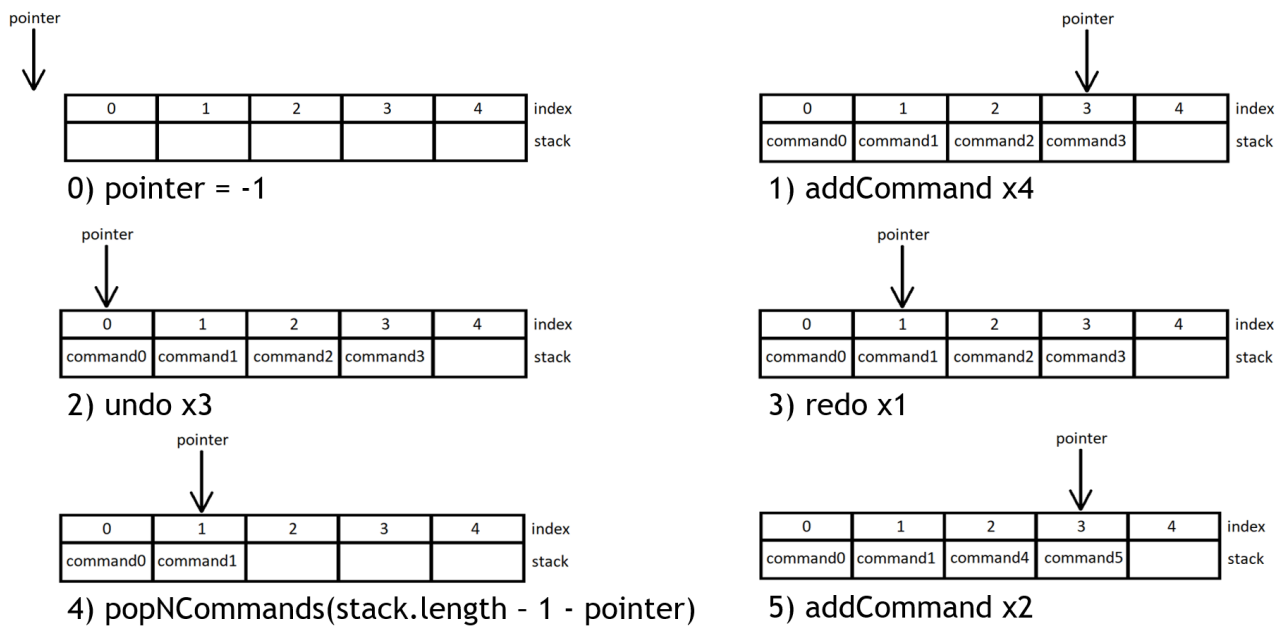


Рис. 4: Пример работы класса UndoRedoController в WMP

1.2 Draw.io

Draw.io [3] – бесплатный веб-сервис, предназначенный для создания различных диаграмм и блок-схем. Инструмент имеет довольно примитивный интерфейс, но благодаря этому приложение не требует особых знаний и усилий для работы в нем. Его редактор был написан на языке JavaScript.

Реализация операций отмены и возврата в данном случае выполнялась следующим образом: аналогично WMP редактор draw.io использует массив для хранения операций и данных, над которыми совершались данные операции. Для контроля над этими составляющими также создается менеджер, управляющий отменой и возвратом.

1.3 Архитектура редактора

Веб-редактор в данный момент выглядит следующим образом: в левой части экрана представлена палитра, в которой есть элементы. Перетаскиванием эти элементы можно переместить на сцену, которая составляет большую часть веб-редактора. Внешний вид веб-редактора можно увидеть на рис. 5.

Что касается архитектуры редактора, то имеется следующее: у класса Editor есть несколько полей, в некоторых из которых хранятся контроллеры

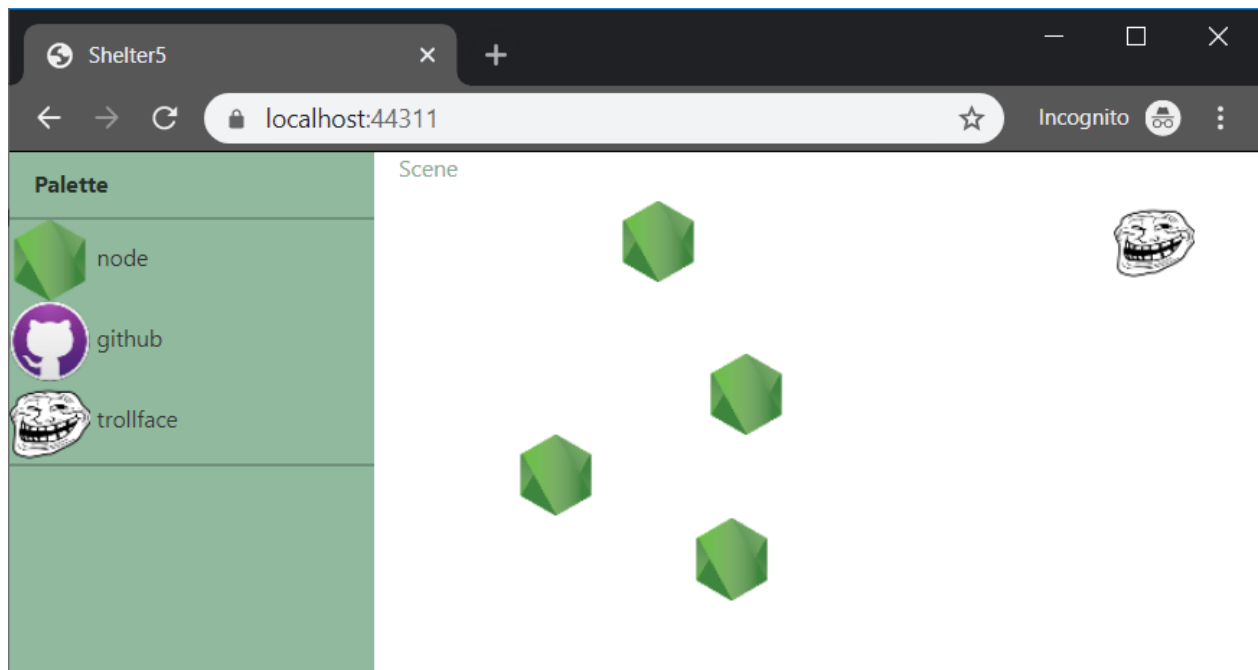


Рис. 5: Редактор

палитры и undo/redo. Класс `PaletteController` управляет палитрой, а класс `UndoRedoController`, в свою очередь, руководит операциями отмены и возврата в редакторе. Графическую интерпретацию архитектуры редактора можно увидеть на рис. 6.

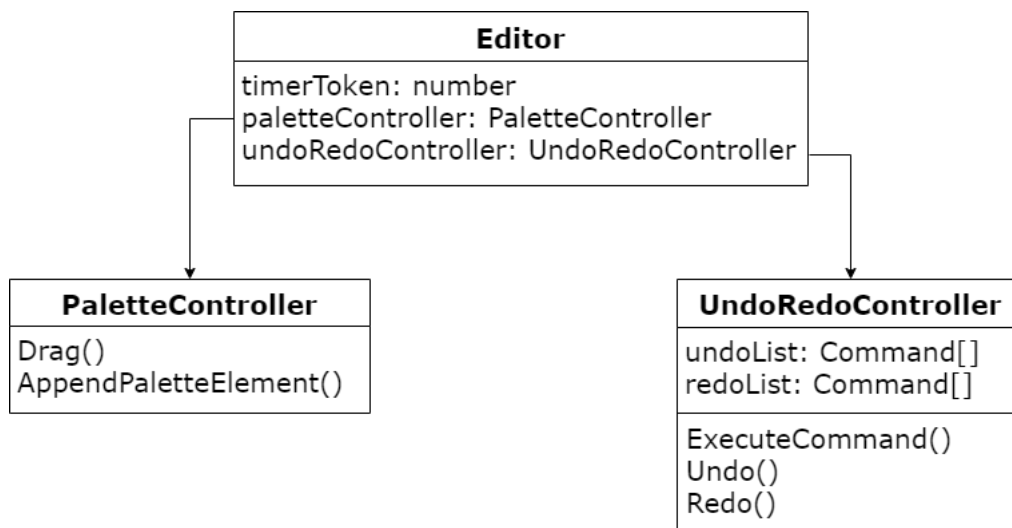


Рис. 6: Архитектура редактора

2. Описание решения

Для разработки веб-приложений раньше широко использовался популярный язык JavaScript, на котором было написано огромное количество различных программ. Однако в последние годы многие программисты переходят на другой язык TypeScript, который был выбран в том числе и для работы над данным проектом, ведь он имеет целый ряд преимуществ.

- Строгая типизация.
- Объектно-ориентированность.
- Перекрестная совместимость с JavaScript.
- Поддержка популярными средами разработки и инструментами.
- Поддержка большого количества различных библиотек.

Для разработки палитры элементов редактора был использован паттерн Model-View-Controller. Использование данного шаблона позволяет модифицировать внутренние и внешние составляющие независимо друг от друга, что является огромным преимуществом при разработке веб-программ в команде. Палитра состоит из трех частей: модели, содержащей данные об элементах палитры и их соотношениях, представления, то есть внешнего вида элементов, и контроллера, который отображает доступные элементы палитры, а также делает их способными к перетаскиванию из палитры на сцену и перемещению по ней.

Палитра была добавлена в разрабатываемый веб-редактор платформы REAL.NET. В палитру были добавлены узлы с названием и изображением, которые можно перетащить на сцену, скопировав их. В реализации палитры была использована библиотека jQuery [4], которая позволяет менять содержимое HTML-документов, взаимодействуя с объектами модели.

Для внедрения поддержки операций отмены и возврата было решено взять за основу широко распространенный паттерн Command, когда каждая конкретная команда является реализацией общего интерфейса. В нашем случае имеется класс AddElementCommand, который является реализацией ин-

терфейса Command. Также существует класс-контроллер UndoRedoController, у которого есть два списка, работающие по принципу стека: undoList и redoList, хранящие команды, которые можно соответственно отменить или вернуть. При выполнении новой команды, эта самая команда помещается в стек undoList. При выполнении операции отмены команды из стека undo перекладываются в стек redo. Аналогичные действия происходят при возврате: команды из стека redo перемещаются в стек undo. Если происходит выполнение новой команды, то redoList очищается. Графическая интерпретация реализации undo/redo в редакторе представлена на рис. 7. Пример работы класса UndoRedoController в редакторе можно увидеть на рис. 8.

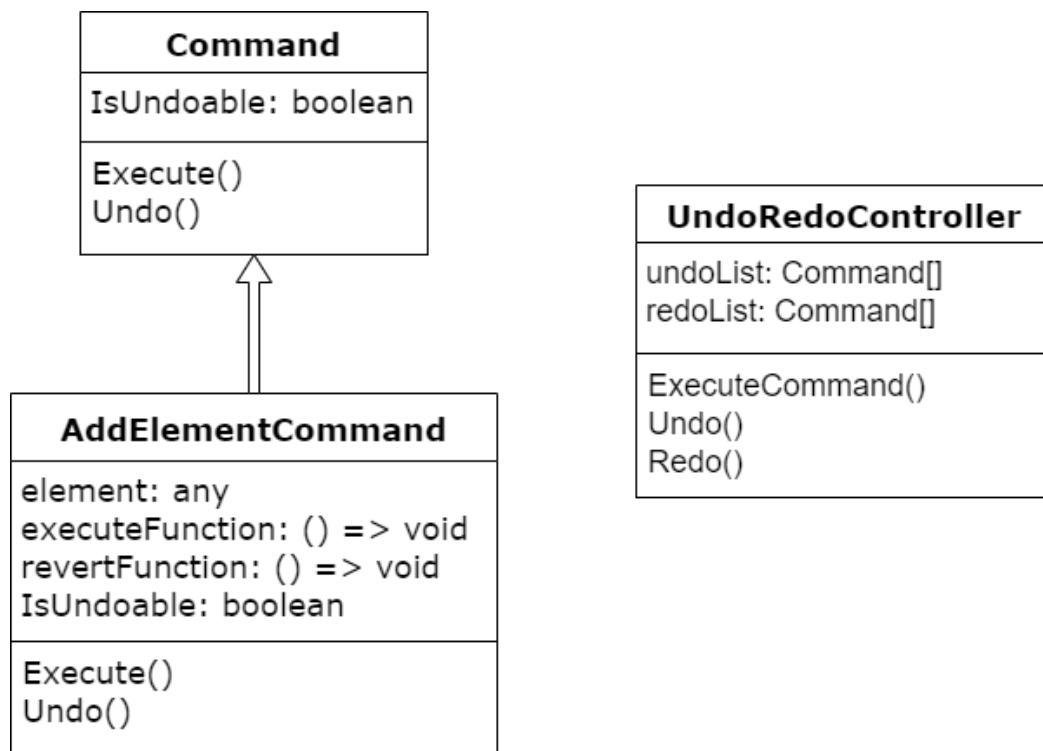


Рис. 7: Undo/Redo в редакторе

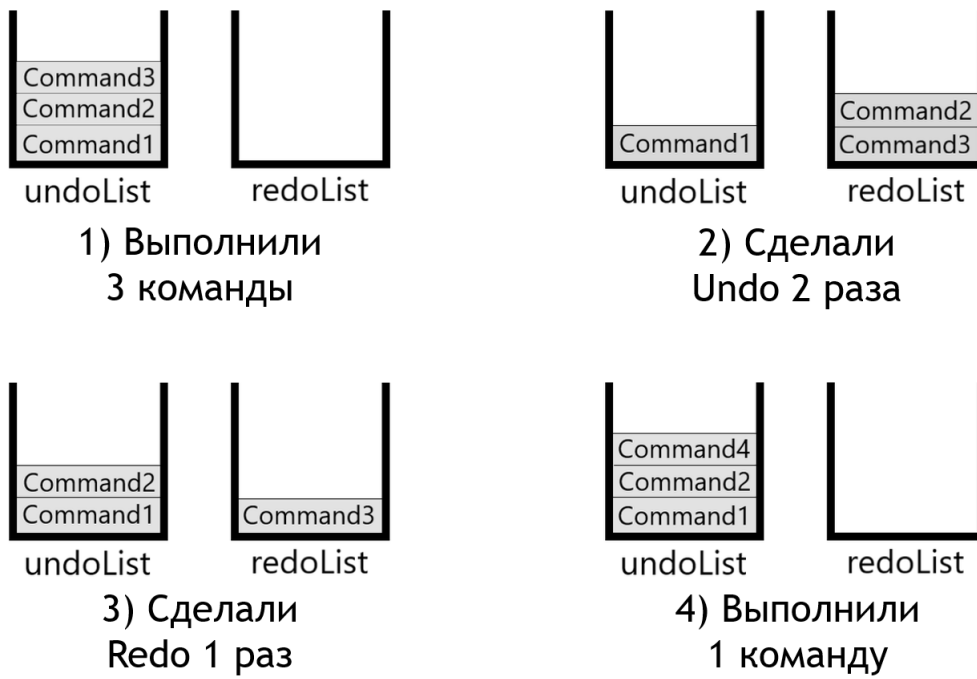


Рис. 8: Пример работы класса `UndoRedoController` в редакторе

Заключение

В рамках данной учебной практики были достигнуты следующие результаты.

- Изучен язык программирования TypeScript.
- Выполнен обзор нескольких существующих веб-редакторов.
- Внедрена поддержка операций отмены и возврата для существующих элементов и операций веб-редактора.
- Реализована примитивная палитра элементов.

Текущие результаты представлены в репозитории на GitHub [5].

Список литературы

- [1] Литвинов Ю.В., Кузьмина Е.В, Небогатилов И.Ю, Алымова Д.А. «Среда предметно-ориентированного программирования REAL.NET», 2017.
URL: <http://se.math.spbu.ru/SE/seminar/materials/realnet.pdf> (дата обращения: 21.12.2019).
- [2] Репозиторий среды разработки «Web Modeling Project».
URL: <https://github.com/qreal/wmp> (дата обращения: 21.12.2019).
- [3] Репозиторий веб-сервиса «draw.io».
URL: <https://github.com/jgraph/drawio> (дата обращения: 21.12.2019).
- [4] Документация jQuery.
URL: <https://api.jquery.com> (дата обращения: 21.12.2019).
- [5] Репозиторий веб-редактора «Multipurpose-editor».
URL: <https://github.com/REAL-NET/Multipurpose-editor> (дата обращения: 21.12.2019).
- [6] Joseph Spinelli. «MVC Overview».
URL: https://medium.com/@joespinelli_6190/mvc-model-view-controller-ef878e2fd6f5 (дата обращения: 21.12.2019).
- [7] Gunnar Peipman. «Composite command pattern».
URL: <https://gunnarpeipman.com/composite-command-pattern> (дата обращения: 21.12.2019).