

СОЗДАНИЕ UML-ДИАГРАММ ПОСЛЕДОВАТЕЛЬНОСТИ: АНАЛИЗ ВЕБ-ИНСТРУМЕНТОВ

Т. В. Митрофанова¹⁾, А. В. Христофорова²⁾, П. А. Шалимова³⁾

¹⁾ Чувашский государственный университет им. И.Н. Ульянова,
Московский пр., 15, г. Чебоксары, Россия, mitrofanova_tv@mail.ru

²⁾ Чувашский государственный университет им. И.Н. Ульянова,
Московский пр., 15, г. Чебоксары, Россия, dlya.nastenki@mail.ru

³⁾ Чебоксарский филиал РАНХиГС,
пр. Максима Горького, 5, 428001, г. Чебоксары, Россия, shalimova-pa@ranepa.ru

В статье рассмотрены инструменты UML для построения диаграмм последовательности PlantUML, SequenceDiagram.org и Mermaid, а также основные различия между ними. Перечисленные в статье веб-инструменты UML позволяют быстро и легко создавать диаграммы последовательности UML с использованием декларативных языков. Текстовая обработка позволяет легко редактировать их, проверять в системе управления версиями и манипулировать ими в существующих IDE.

Ключевые слова: диаграмма последовательности; UML; моделирование; информационная система; PlantUML; SequenceDiagram.org; Mermaid.

UML – это универсальный визуальный язык, который используется для моделирования программной системы [1]. В индустрии программного обеспечения мнения относительно использования UML-диаграмм разделились. В то время как некоторые рассматривают его как неотъемлемую часть программных систем и разработки, есть значительное количество людей, которые считают его совершенно ненужным.

Студенты Чувашского государственного университета на 3 курсе знакомятся с данным языком графического описания для объектного моделирования в области разработки программного обеспечения в рамках дисциплины «Проектирование информационно-вычислительных систем». Для обучения построению диаграмм было принято решение использовать веб-инструменты за счет их доступности и простоты использования.

Хотя существует 14 различных типов UML-диаграмм для моделирования приложений, используется только три или четыре для документирования программной системы. Диаграммы классов, диаграммы последовательностей и диаграммы вариантов использования остаются наиболее популярными. Из них диаграммы последовательности являются одними из наиболее часто используемых в современной программной инженерии и чаще всего вызывает затруднения [2].

Разнообразие инструментов UML – одна из важнейших причин, по которой UML так широко используется. Существует множество универсальных инструментов для создания диаграмм, которые можно использовать для разработки программного обеспечения: Diagrams.net, Visual Paradigm и др. Эти универсальные инструменты обеспечивают большую гибкость, но в конечном счёте тратится больше времени на перетаскивание фигур, изменение размера, копирование и вставку, перетаскивание стрелок, их перестановку, а также перестановку всех элементов, потому что они больше не помещаются на холсте.

Подход, набирающий популярность, заключается в использовании инструментов, которые позволяют использовать «диаграммы как код» [4]. Таким образом, диаграмма описывается с помощью кода, а инструменты заботятся о визуальном рендеринге. Дополнительным преимуществом является то, что вы можете управлять исходным кодом своих диаграмм и периодически проверять изменения.

Онлайн-редакторы UML:

- Kkeisuke редактор (<https://plantuml-editor.kkeisuke.com/>);
- Planttext редактор (<https://www.planttext.com/>);

Оригинальный редактор

(<http://www.plantuml.com/plantuml/uml/SyfFKj2rKt3CoKnELR1Io4ZDoSa70000>) и др.

Диаграмма последовательности (sequence diagram) – диаграмма, на которой показаны взаимодействия объектов, упорядоченные по времени их проявления [3].

Диаграммы последовательностей эффективны для:

- более подробного объяснения сценариев использования перед началом кодирования;
- предоставления четких рекомендаций разработчикам;
- моделирования сложной операции, функции или рабочего процесса;
- демонстрации объектов, компонентов и их взаимосвязей в последовательности.

Рассмотрим диаграммы последовательности в различных популярных инструментах типа «диаграммы как код».

PlantUML является бесплатным, с открытым исходным кодом (на основе Java) по GNU General Public License.

Характеристики:

- создание графических изображений в формате ASCII для диаграмм последовательностей;
- пользовательский стиль с использованием таблиц стилей CSS;
- обширная настройка наконечников стрелок, плавных линий, действующих лиц и значков;
- экспорт в PNG, SVG, EPS, PDF, VDX, XMI, TXT, Latex и HTML.

На рис. 1 представлена схема взаимодействия в оригинальном редакторе PlantUML.

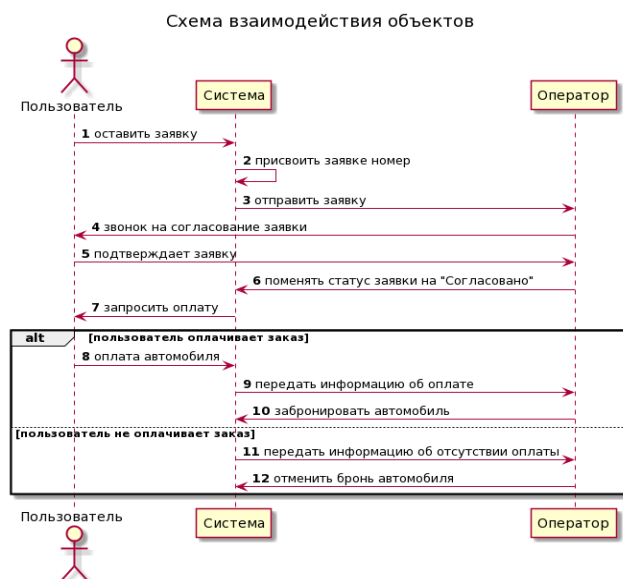


Рис. 1. Схема взаимодействия в PlantUML

Скрипт PlantUML для этой диаграммы:

```

@startuml
autonumber
title Схема взаимодействия объектов
actor Пользователь
participant Система
participant Оператор
Пользователь->>Система: оставить заявку
  
```

```

Система-> Система: присвоить заявке номер
Система-> Оператор: отправить заявку
Оператор-> Пользователь: звонок на согласование заявки
Пользователь-> Оператор: подтверждает заявку
Оператор-> Система: поменять статус заявки на "Согласовано"
Система-> Пользователь: запросить оплату
alt пользователь оплачивает заказ
Пользователь-> Система: оплата автомобиля
Система-> Оператор: передать информацию об оплате
Оператор-> Система: забронировать автомобиль
else пользователь не оплачивает заказ
Система-> Оператор: передать информацию об отсутствии оплаты
Оператор-> Система: отменить бронь автомобиля
end
@enduml

```

Рассмотрим онлайн-инструмент для построения диаграмм последовательностей – сайт SequenceDiagram.org (<https://sequencediagram.org/>). Для него характерны понятный простой интерфейс и быстрая обработка на стороне клиента.

Инструмент является бесплатным для базового проектирования диаграмм последовательности и платным для таких функций, как история изменений, брендинг, параллельные потоки и совместная работа.

Характеристики:

- множество предустановленных вариантов оформления на выбор;
- автоматическое сохранение и история изменений;
- возможность включать и повторно использовать текст из других диаграмм;
- REST API для плагинов и расширений;
- экспорт в PNG, PDF, SVG, текст и URL.

На рис. 2 представлена схема взаимодействия в инструменте SequenceDiagram.org.

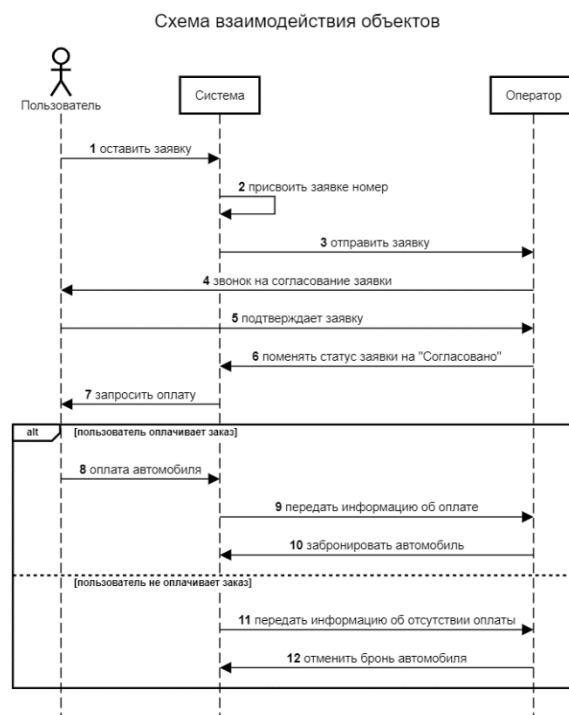


Рис. 2. Схема взаимодействия в SequenceDiagram.org

Для диаграмм последовательности синтаксис PlantUML и SequenceDiagram.org аналогичны, визуализированное изображение незначительно отличается.

Рассмотрим еще один инструмент для построения диаграмм на основе JavaScript, который использует текстовые определения в стиле Markdown и средство визуализации для создания и изменения сложных диаграмм – Mermaid.

Mermaid является бесплатным, с открытым исходным кодом по лицензии MIT. Его диаграммы GitHub отобразит в файлах Markdown. Mermaid поддерживается сообществом разработчиков с открытым исходным кодом, он поддерживает сообщения, заметки, альты, циклы, параллели, разрывы и подсветку фона, а также выполнит настройку и стилизацию с использованием таблиц стилей CSS.

На рис. 3 представлена схема взаимодействия в Mermaid (для работы онлайн используется mermaid.live).

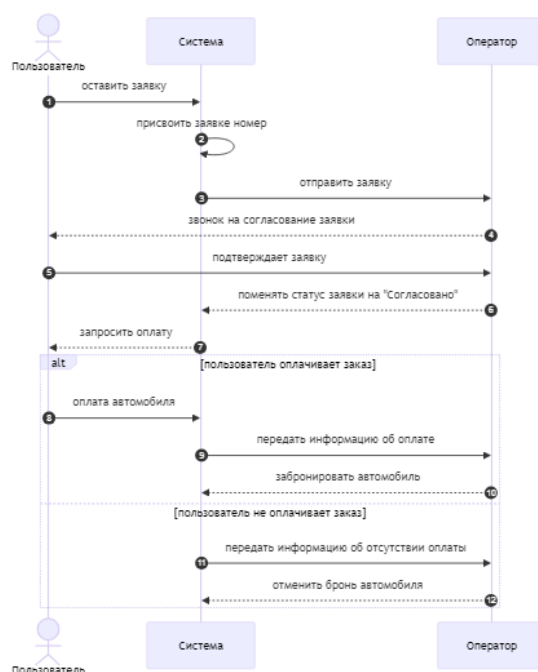


Рис. 3. Схема взаимодействия в Mermaid

Скрипт Mermaid для этой диаграммы:

sequenceDiagram

autonumber

actor Пользователь

participant Система

participantОператор

Пользователь->>Система:оставить заявку

Система->>Система:присвоить заявке номер

Система->>Оператор:отправить заявку

Оператор->>Пользователь:звонок на согласование заявки

Пользователь->>Оператор:подтверждает заявку

Оператор->>Система:поменять статус заявки на "Согласовано"

Система->>Пользователь:запросить оплату

Alt пользователь оплачивает заказ

Пользователь->>Система:оплата автомобиля

```

Система->>Оператор:передать информацию об оплате
Оператор-->>Система:забронировать автомобиль
else пользователь не оплачивает заказ
Система->>Оператор:передать информацию об отсутствии оплаты
Оператор-->>Система:отменить бронь автомобиля
end

```

Синтаксис немного отличается от PlantUML, SequenceDiagram.org, визуализированное изображение отличается незначительно.

Таким образом, мы рассмотрели инструменты UML для построения диаграмм последовательности: PlantUML, SequenceDiagram.org и Mermaid, а также основные различия между ними. Большая часть синтаксиса у них схожа, но Mermaid, как правило, немного лучше справляется с визуализацией более привлекательной диаграммы.

Библиографические ссылки

1. Il-Yeol S. Developing Sequence Diagrams in UML. In: S.Kunii H., Jajodia, S. Sølvsberg A. (eds) Conceptual Modeling ER 2001 [Electronic resource]// Lecture Notes in Computer Science, vol 2224. Springer, Berlin, Heidelberg. URL: https://doi.org/10.1007/3-540-45581-7_28.
2. Вичугова А. Как построить UML-диаграмму последовательности: практический пример [Электронный ресурс] // BABOKSCHOOL. Школа прикладного бизнес-анализа. URL: <https://babok-school.ru/blogs/uml-sequence-diagram-example/>
3. Диаграмма последовательности (sequencediagrams) [Электронный ресурс] // Техноблог. О технологиях, бизнесе и интернете/. URL: https://itonboard.ru/analysis/394-diagramma_posledovatelnosti_sequence_diagrams_uml/.
4. Овзьяк А. Plantuml в работе системного аналитика. Пиши uml диаграммы текстом, чтобы сэкономить время [Электронный ресурс] // Хабр. URL: <https://habr.com/ru/articles/661931/>