



UML для разработки

Кошкин В.А.

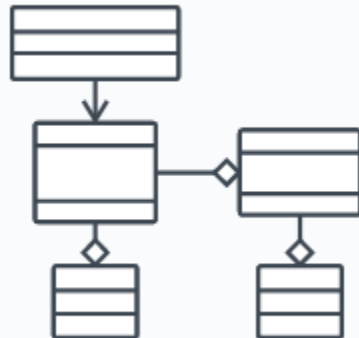
Что такое UML?

Унифицированный язык моделирования (UML) был разработан с целью обеспечить единый визуальный язык с богатой семантикой и развернутым синтаксисом для проектирования и внедрения программных систем со сложной структурой и комплексным поведением. Стоит отметить, что UML применяется не только в разработке программ, но и в других сферах, например, в схематизации потоков производственных процессов.



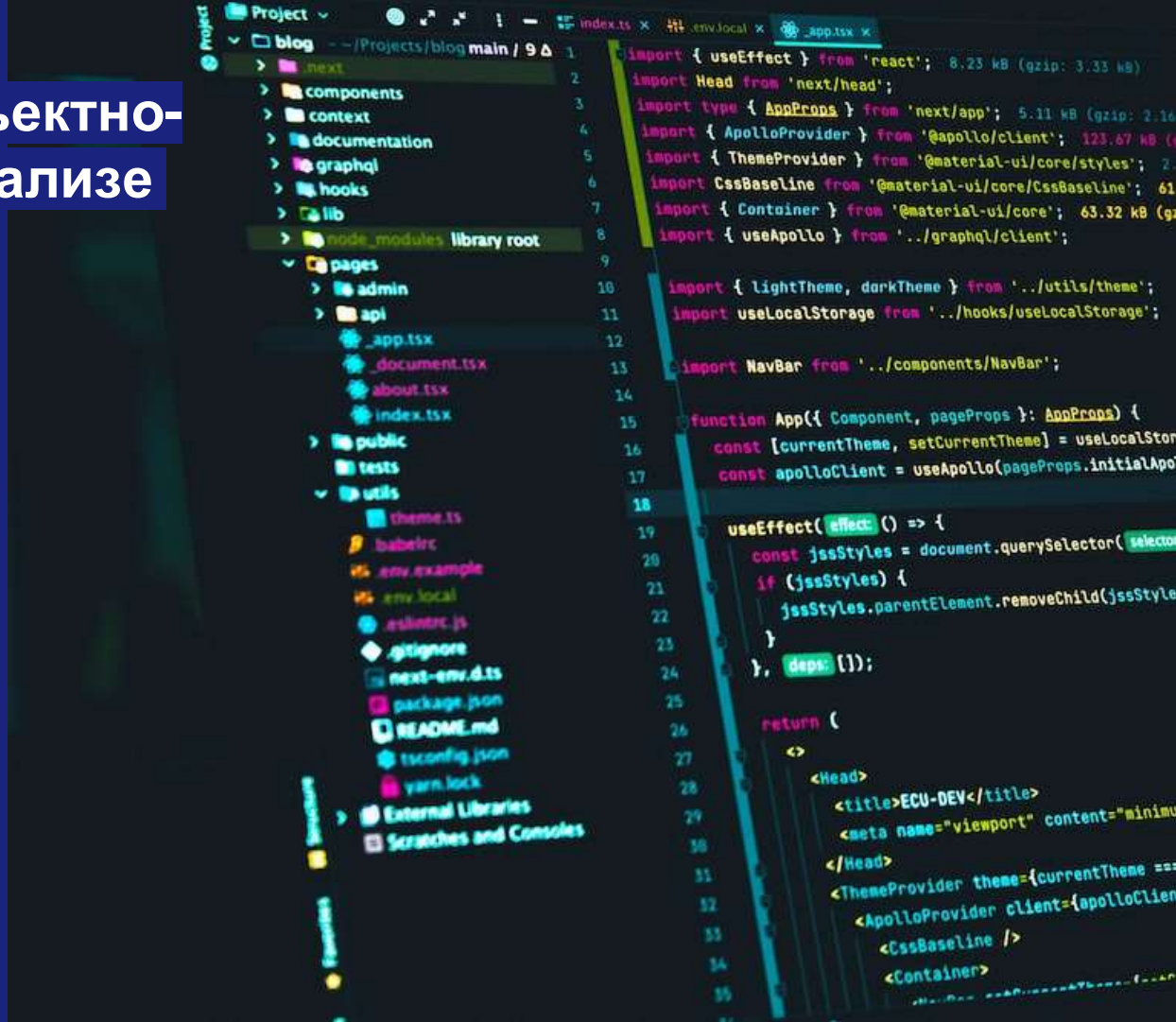
UML для генерации

UML не является языком программирования, однако на базе диаграмм UML можно сгенерировать код на разных языках, и для этого существует ряд специальных инструментов. Зато с объектно-ориентированным анализом и дизайном унифицированный язык моделирования связан напрямую.



UML и его роль в объектно-ориентированном анализе и дизайне

Объектно-ориентированные языки преобладают в сфере программирования, так как моделируют объекты реального мира. UML сочетает в себе несколько разновидностей объектно-ориентированной нотации — объектно-ориентированный дизайн, технику объектного моделирования и разработку объектно-ориентированного программного обеспечения.



UML и моделирование данных

Язык UML получил широкое распространение в среде программистов, однако в целом мало используется в разработке баз данных. Одна из причин этого заключается в том, что создатели UML не ставили базы данных в центр внимания. Тем не менее, UML эффективно применяется в концептуальном моделировании данных высоких уровней и подходит для создания различных видов диаграмм UML.

Типы UML диаграмм

В языке UML есть 12 типов диаграмм:

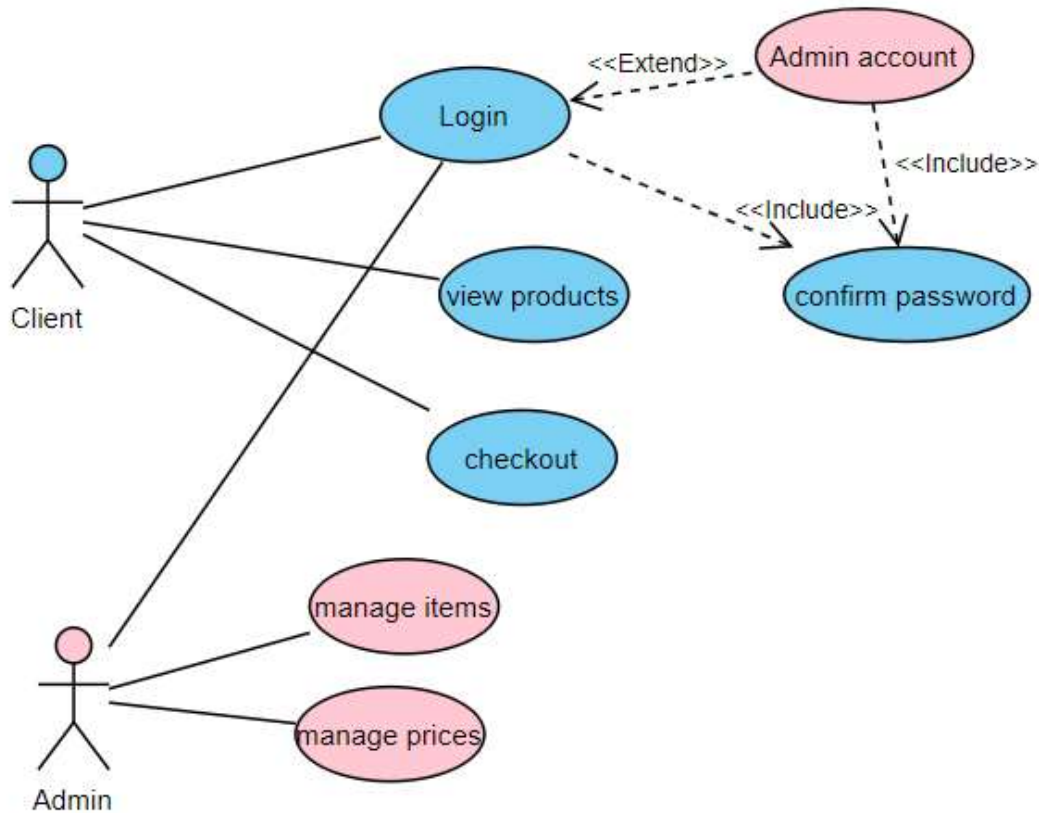
- 4 типа диаграмм представляют **статическую структуру** приложения;
- 5 типов представляют **поведенческие аспекты** системы;
- 3 представляют **физические аспекты** функционирования системы (диаграммы реализации).

Типы UML диаграмм

Некоторые из видов диаграмм специфичны для определенной системы и приложения. Самыми доступными из них являются:

- Диаграмма прецедентов (Use-case diagram);
- Диаграмма классов (Class diagram);
- Диаграмма активностей (Activity diagram);
- Диаграмма последовательности (Sequence diagram);
- Диаграмма развёртывания (Deployment diagram);
- Диаграмма сотрудничества (Collaboration diagram);
- Диаграмма объектов (Object diagram);
- Диаграмма состояний (Statechart diagram).





1) Actor (участник) — множество логически связанных ролей, исполняемых при взаимодействии с прецедентами или сущностями. Участником может быть человек, роль человека в системе или другая система, подсистема или класс, которые представляют нечто вне сущности.

2) Use case (прецедент) — описание отдельного аспекта поведения системы с точки зрения пользователя. Прецедент не показывает, "как" достигается некоторый результат, а только "что" именно выполняется.

ДИАГРАММА КЛАССОВ(CLASS DIAGRAM)

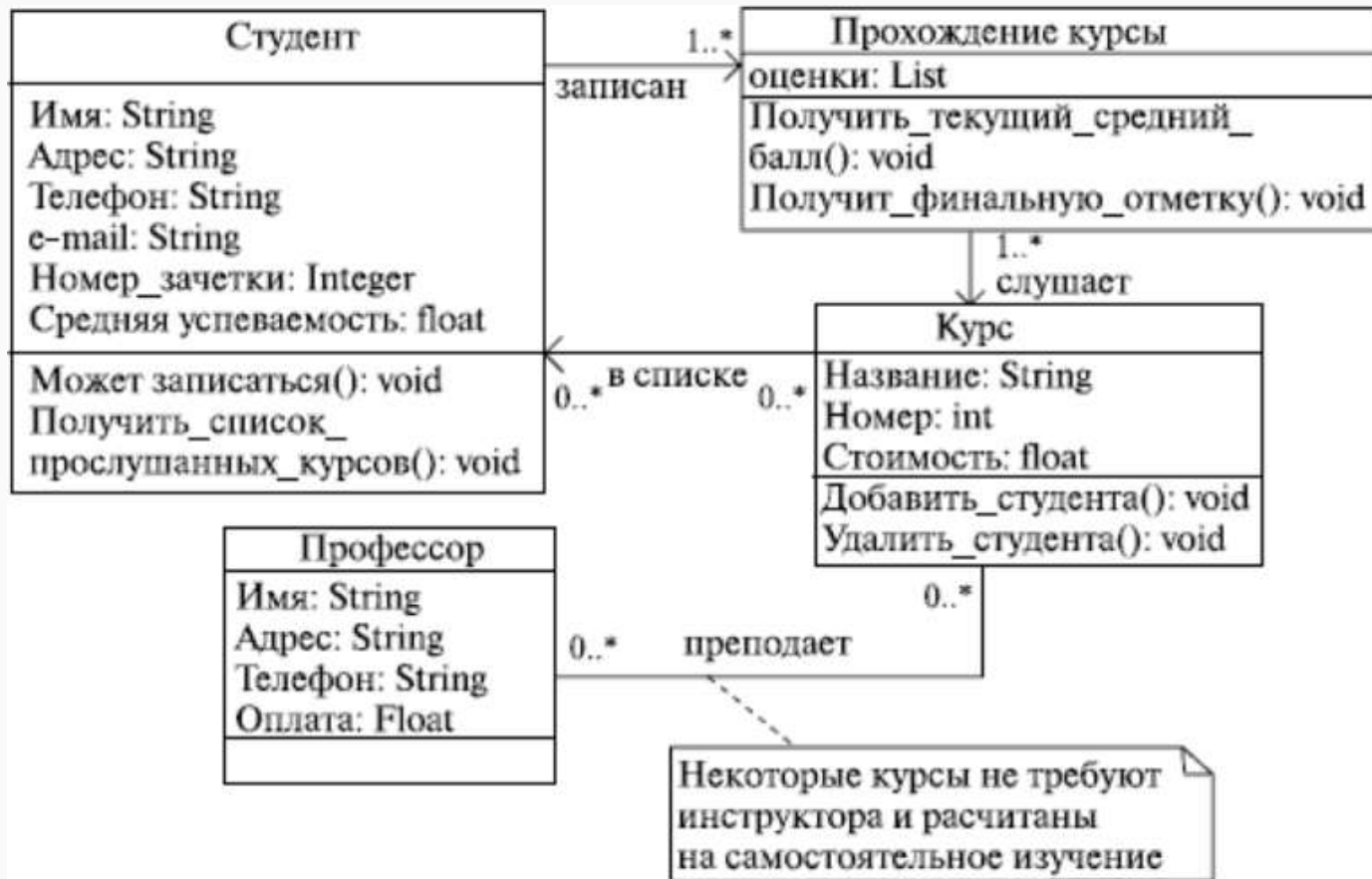


ДИАГРАММА АКТИВНОСТЕЙ (ACTIVITY DIAGRAM)

Диаграмма активностей описывает динамические аспекты поведения системы в виде блок-схемы, которая отражает бизнес-процессы, логику процедур и потоки работ — переходы от одной деятельности к другой. По сути, мы рисуем алгоритм действий (логику поведения) системы или взаимодействия нескольких систем.

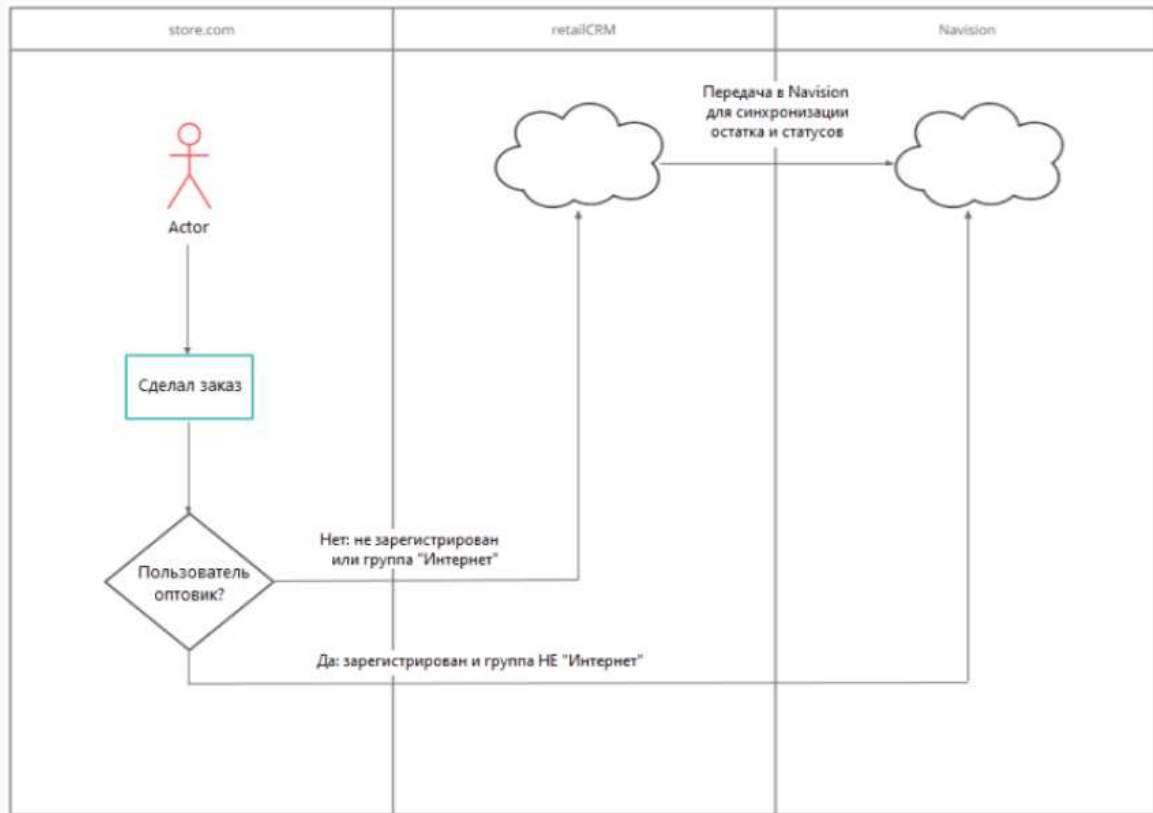


ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТЕЙ И (SEQUENCE DIAGRAM)

Используется для уточнения диаграмм прецедентов — описывает поведенческие аспекты системы. Диаграмма последовательности отражает взаимодействие объектов в динамике, во времени. При этом информация принимает вид сообщений, а взаимодействие объектов подразумевает обмен этими сообщениями в рамках сценария.

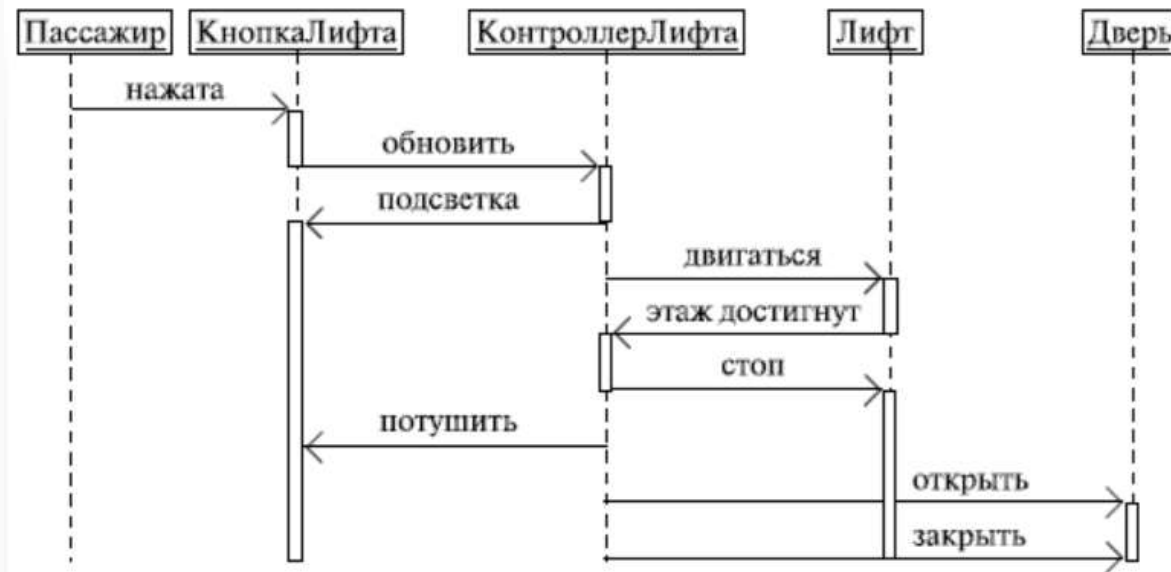


ДИАГРАММА РАЗВЕРТЫВАНИЯ(DEPLOYMENT DIAGRAM)



Диаграмма развертывания отображает графическое представление инфраструктуры, на которую будет развернуто приложение: топологию системы и распределение компонентов по ее узлам, а также соединения — маршруты передачи данных между узлами. Диаграмма помогает более рационально организовать компоненты, от чего зависит в числе прочего и производительность системы, а также решить вспомогательные задачи, например, связанные с безопасностью.



UML для разработки

Кошкин В.А.