



BABOKSCHOOL
ШКОЛА ПРИКЛАДНОГО БИЗНЕС-АНАЛИЗА

Systems • Education

- Диаграмма последовательности

ОСНОВЫ UML



Вичугова Анна



- ✓ Работаю в консалтинге по системному и бизнес-анализу с 2009 года
- ✓ Аналитик и технический писатель в Школе Больших Данных <https://www.bigdataschool.ru/>
- ✓ Основатель Школы прикладного бизнес-анализа <https://babok-school.ru/>, автор и преподаватель курсов
- ✓ Ведущий разработчик в Школе системного анализа <https://systems.education/>
- ✓ Кандидат технических наук (Системный анализ, управление и обработка информации, 2013)
- ✓ сертифицированный бизнес-аналитик (ex-SBAP 2020, международная сертификация IIBA® по BABOK®Guide)
- ✓ сертифицированный специалист Business Studio (2010, 2012, 2013, 2018)
- ✓ сертифицированный специалист и администратор СЭД Directum (2011)

Профессиональные интересы: системный анализ, бизнес-анализ, разработка и поддержка СМК, ССП (KPI), анализ и формализация бизнес-процессов и оргструктур, Data Science и Machine Learning, инженерия Big Data, разработка программной документации



Что рассмотрим сегодня?

- **Что такое UML**
 - Краткая история появления
 - Области применения
 - Виды UML-диаграмм
- **Что такое диаграмма последовательности UML?**
 - Назначение
 - Алфавит нотации
 - Кейсы использования для анализа
- **Примеры использования**
 1. Аутентификация пользователя
 2. добавление товаров менеджером (с аутентификацией)
 3. добавление товара в заказ
 4. покупка товара
 5. Асинхронное взаимодействие нескольких систем через Kafka





Что такое UML: краткий ликбез

Версия	Дата принятия
1.1	ноябрь 1997
1.3	март 2000
1.4	сентябрь 2001
1.4.2	июль 2004
1.5	март 2003
2.0	июль 2005
2.1	формально не была принята
2.1.1	август 2007
2.1.2	ноябрь 2007
2.2	февраль 2009
2.3	май 2010
2.4 beta 2	март 2011
2.5	июнь 2015
2.5.1	декабрь 2017

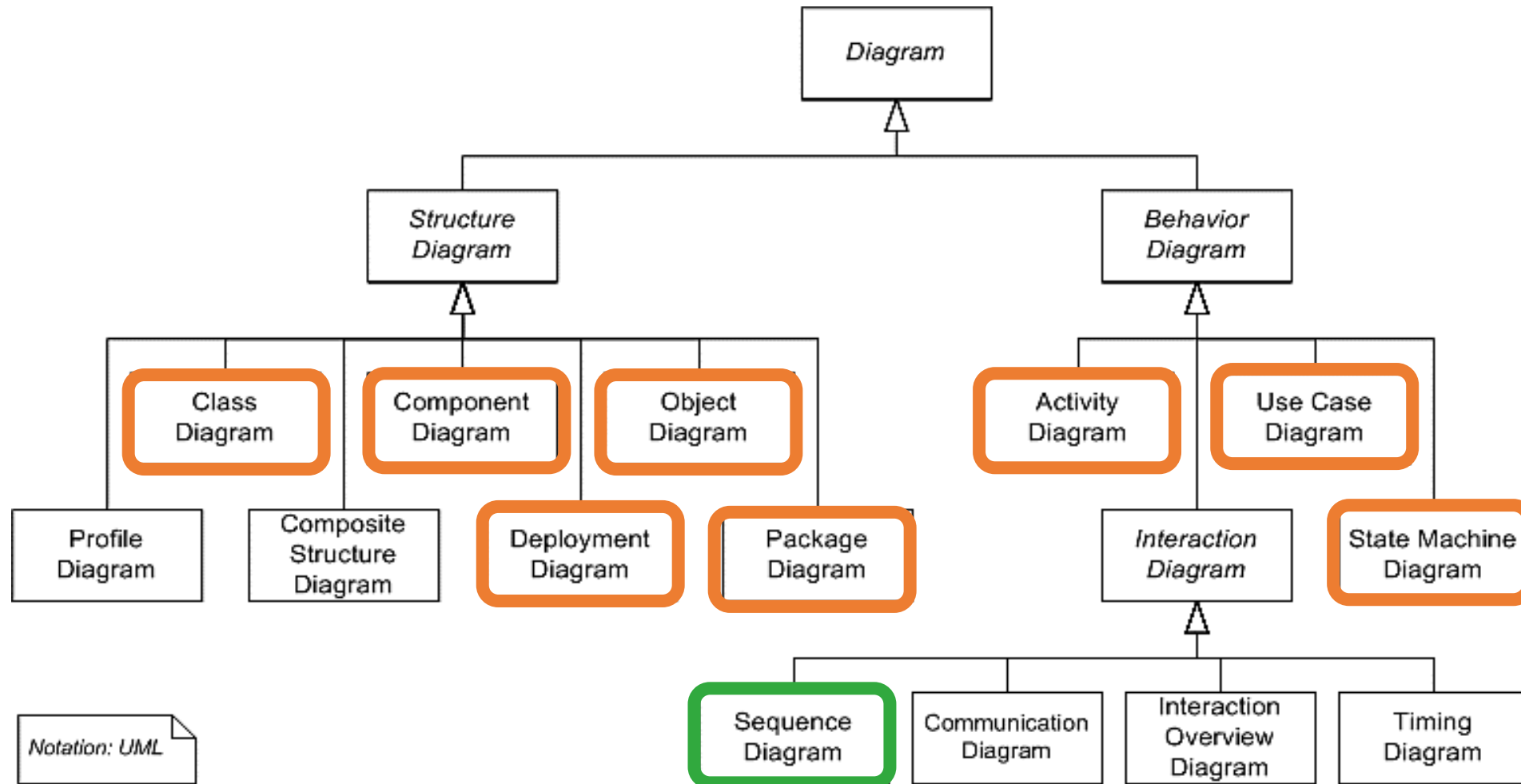
UML (Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения, для моделирования бизнес-процессов, системного проектирования и отображения организационных структур

UML - открытый стандарт, использующий графические обозначения для создания абстрактной модели системы (UML-модели). Впервые UML был создан в 1994 году (OMG, Буч, Рамбо и Якобсон) для определения, визуализации, проектирования и документирования, в основном, программных систем.

UML — это НЕ язык программирования, но UML-модели можно конвертировать в код.



Виды UML-диаграмм





UML-диаграмма последовательности

Activation bar - действие, выполняемое в определенный период на временной шкале объекта.

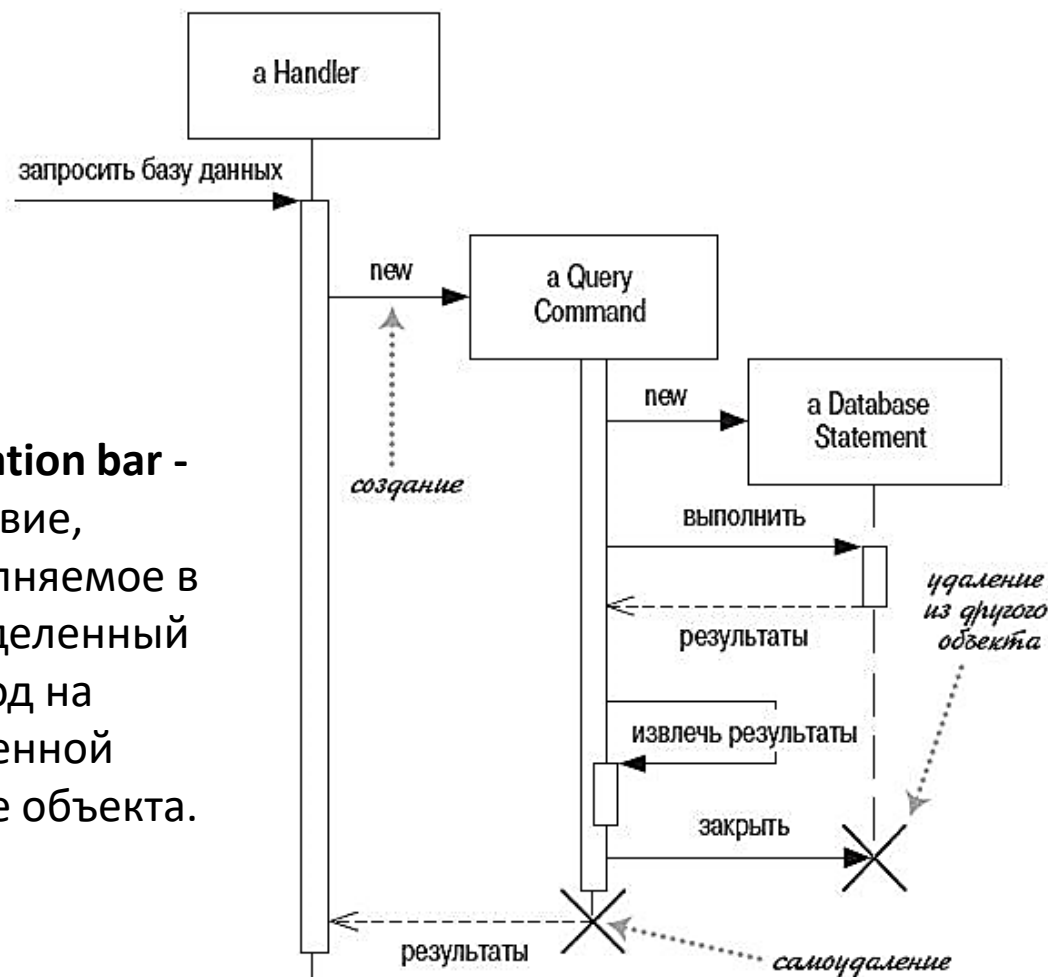


Диаграмма последовательности (Sequence diagram) используется, когда нужно описать поведение нескольких объектов в рамках одного прецедента.

Диаграммы последовательности подходят для представления взаимодействия объектов, но не для точного определения поведения.

Для циклов и условий в диаграммах последовательностей используются фреймы взаимодействия



UML sequence

<https://medium.com/geekculture/system-design-uml-sequence-diagram-5b75fae399e3>

- ✓ Это основанное на взаимодействии моделирование поведения объектов, которое показывает динамическое сотрудничество между несколькими объектами, описывая хронологический порядок, в котором сообщения передаются между ними.
- ✓ Диаграмма последовательности представляет собой двумерную диаграмму.
- ✓ Горизонтальная ось представляет объекты, а вертикальная ось представляет время.
- ✓ Сообщения передаются горизонтально между объектами и располагаются вертикально в хронологическом порядке.
- ✓ Показывается последовательность взаимодействий между объектами: как объекты отправляют и получают сообщения.
- ✓ Можно описать параллельные процессы и временную последовательность взаимодействий

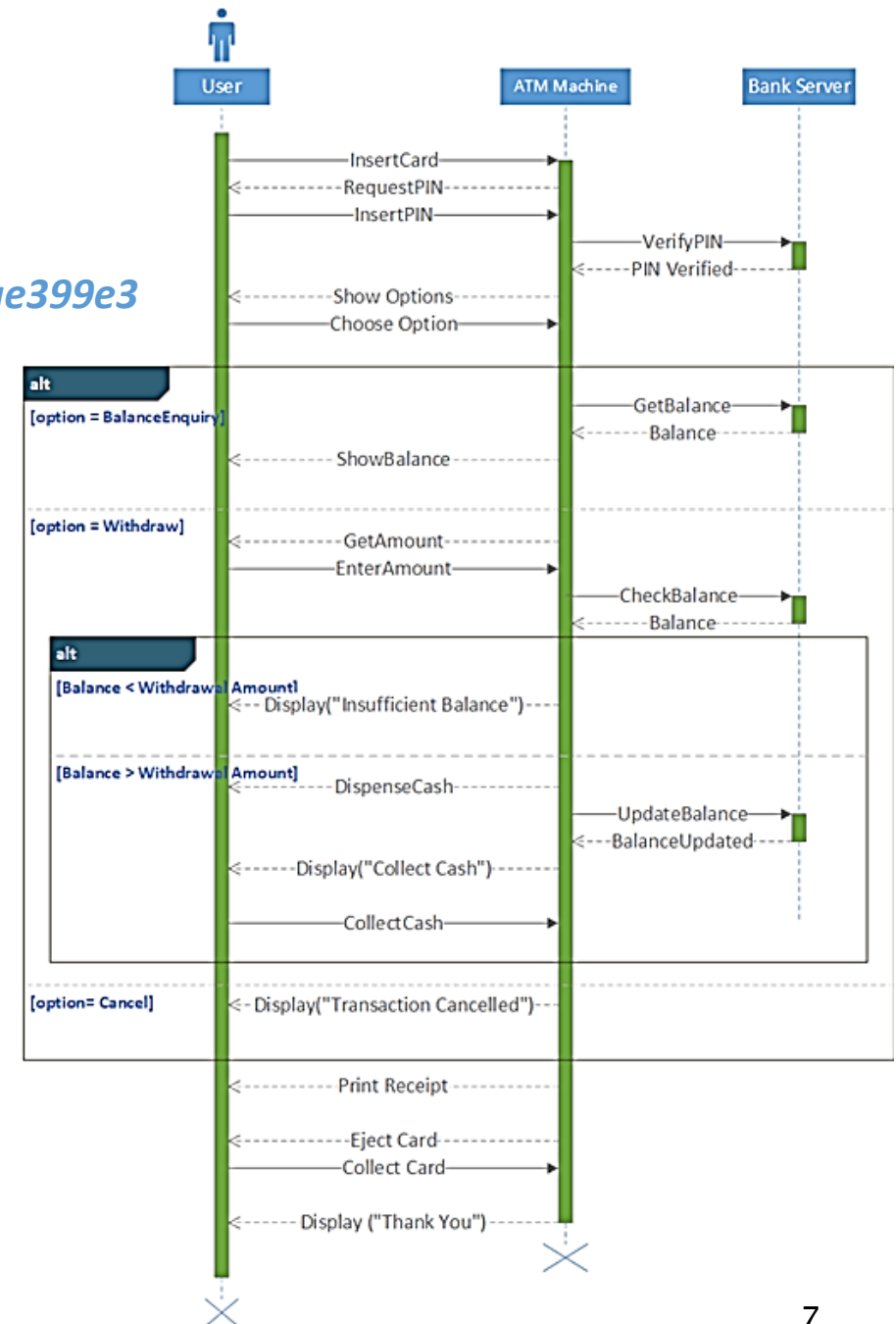




Диаграмма последовательности: виды стрелок

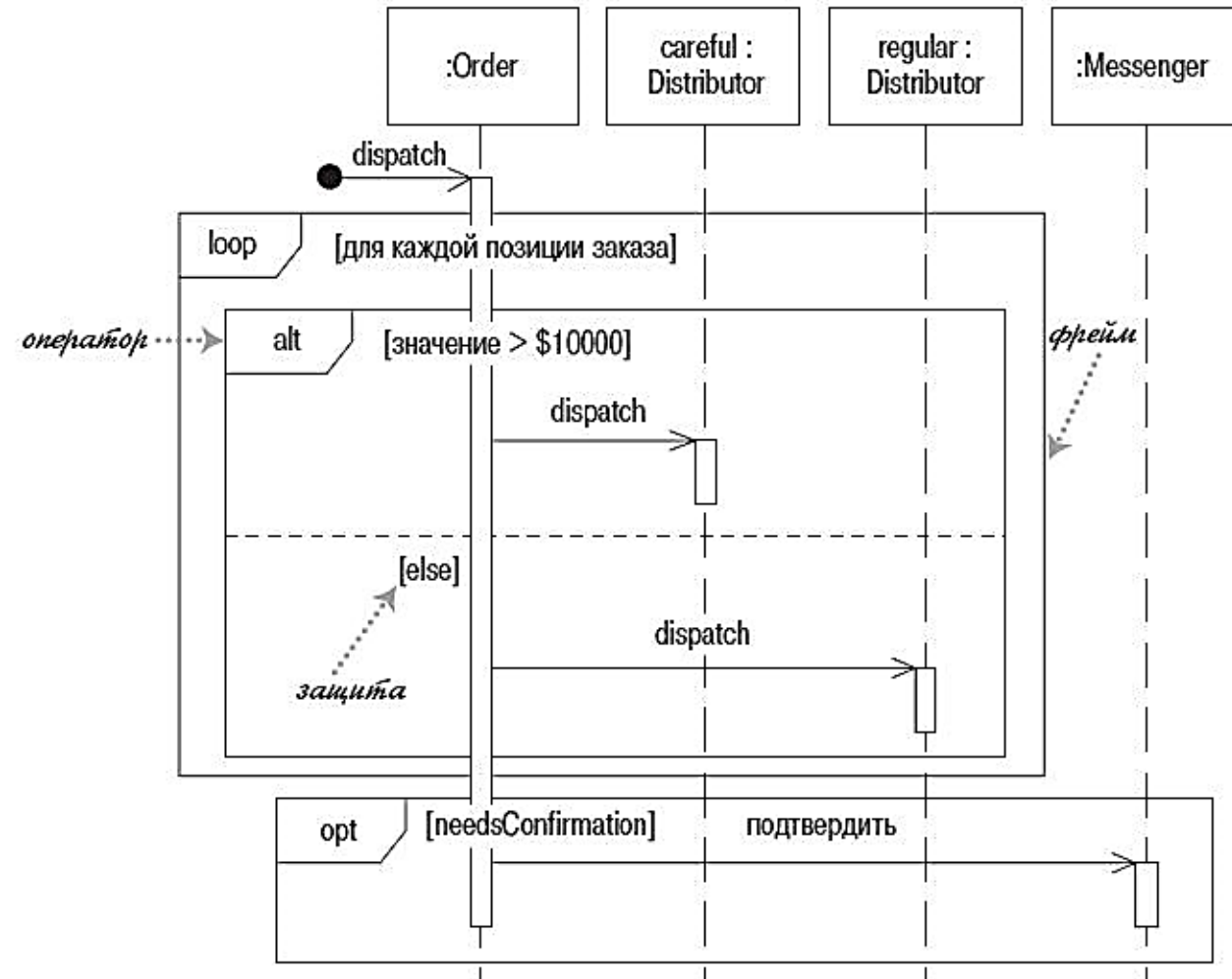
Сообщение	Смысл	Знак
Синхронное	Отправитель передаёт ход управления получателю, который должен выполнить в прецеденте некоторое действие. Пока оно не будет завершено, т.е. не вернется ответное сообщение, отправитель НЕ может выполнять действия.	
Ответное	Ответ на синхронное сообщение - содержит возвращаемое изначальному актору-отправителю значение и возвращает ему управление (возможность действовать).	
Асинхронное	Отправитель передаёт ход управления получателю, который должен выполнить в прецеденте некоторое действие, но при этом отправитель МОЖЕТ совершать другие действия.	
Потерянное	Сообщение без адресата, когда есть отправитель, но нет получателя, он находится за границами проектируемой системы. Используется для межсистемного (интеграционного) взаимодействия.	
Найденное	Сообщение без отправителя, когда есть получатель, но нет отправителя, т.е. он находится за границами проектируемой системы. Используется для межсистемного (интеграционного) взаимодействия.	



UML-диаграмма последовательности

http://www.tracemodeler.com/articles/a_quick_introduction_to_uml_sequence_diagrams/

<https://medium.com/geekculture/system-design-uml-sequence-diagram-5b75fae399e3>



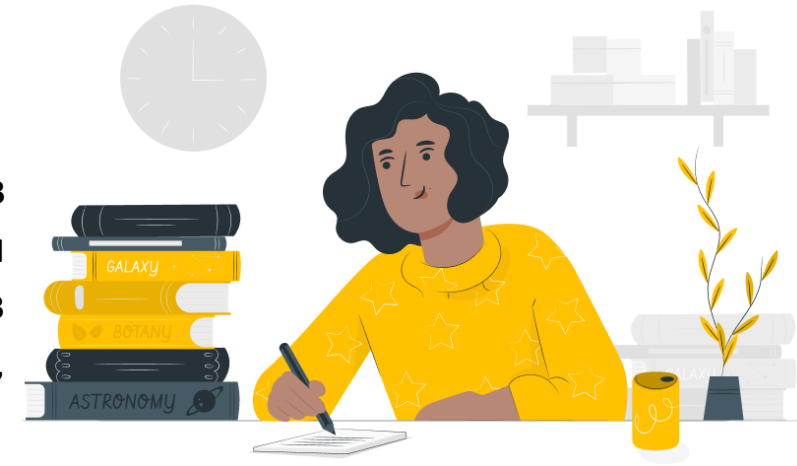
Для циклов и условий в диаграммах последовательности используются фреймы взаимодействия с операторами:

- ✓ **alt** – несколько альтернатив, выполняется только та, для которой условие истинно
- ✓ **opt** – опционально
- ✓ **par** – распараллеливание фрагментов
- ✓ **loop** – цикл, может повторяться несколько раз
- ✓ **region** или **critical** – критическая область с одним потоком, который выполняется за один прием
- ✓ **neg** – неверное взаимодействие
- ✓ **ref** – ссылка на другие диаграммы
- ✓ **sd** – граница sequence diagram



Кейсы

- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.





Служба доставки

Фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.



Задачи на проектирование UML-sequence:

1. отправка заказа на доставку
2. прием заказа
3. расчет стоимости и длительности доставки (с поиском курьеров)
4. обновление истории доставок (у Курьера и Клиента)



Туроператор

Клиенты бронируют проживание в гостинице на сколько-то суток.
Оплата брони может быть онлайн или на месте.

Задачи на проектирование UML-sequence:

1. поиск гостиницы
2. бронирование гостиницы
3. оплата проживания
4. отказ от брони





Такси

Клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.



Задачи на проектирование UML-sequence:

1. заказ такси
2. прием заказа (поиск водителя)
3. расчет стоимости поездки
4. обновление истории поездок (у Водителя и у Клиента)



Банк

Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт предусматривает разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.

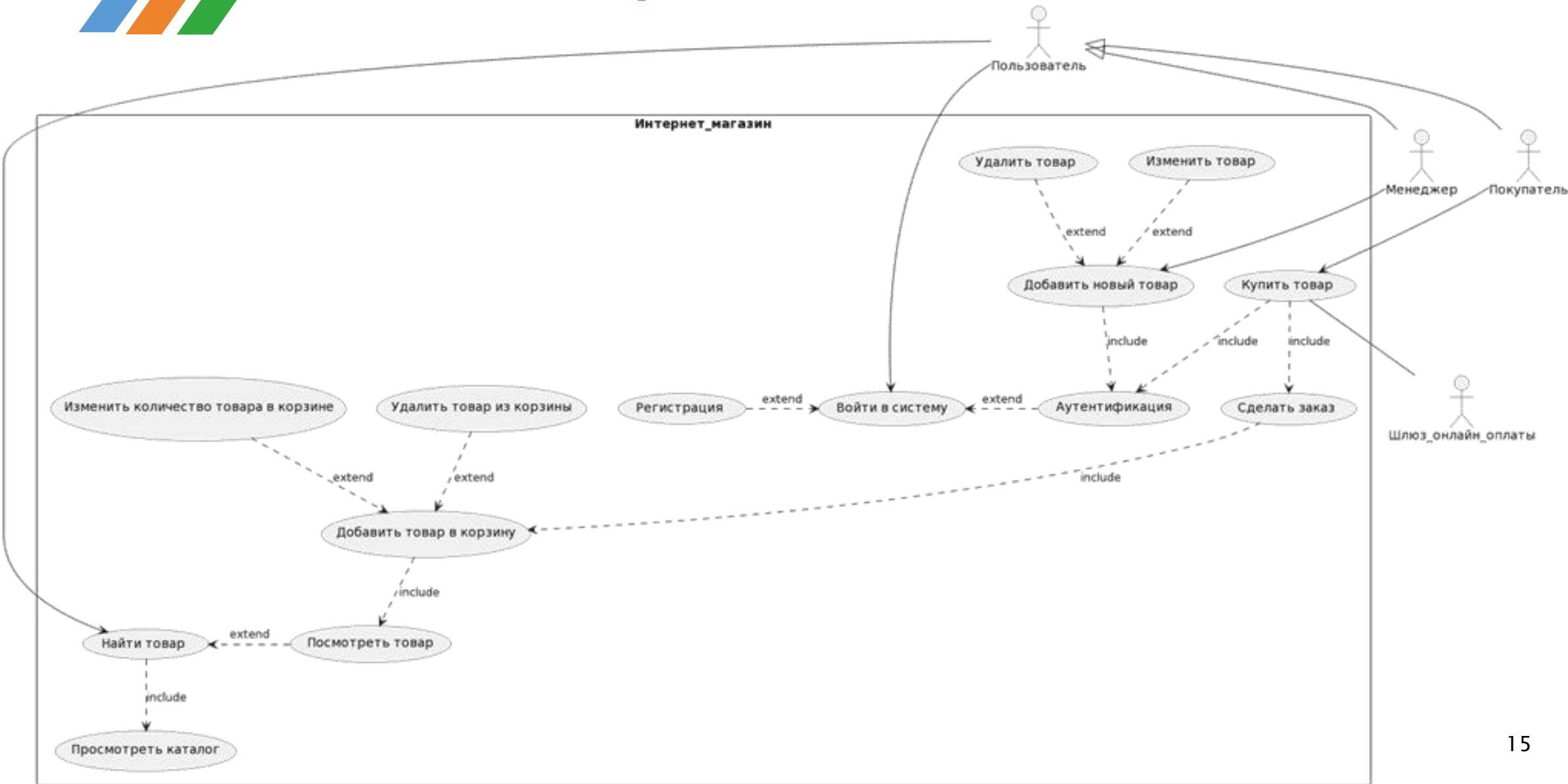


Задачи на проектирование UML-sequence:

1. отправка заявки на кредит
2. скоринг (вычисление оценки) клиентской заявки
3. ручное оценивание клиентской заявки Менеджером
4. отправка уведомлений клиенту



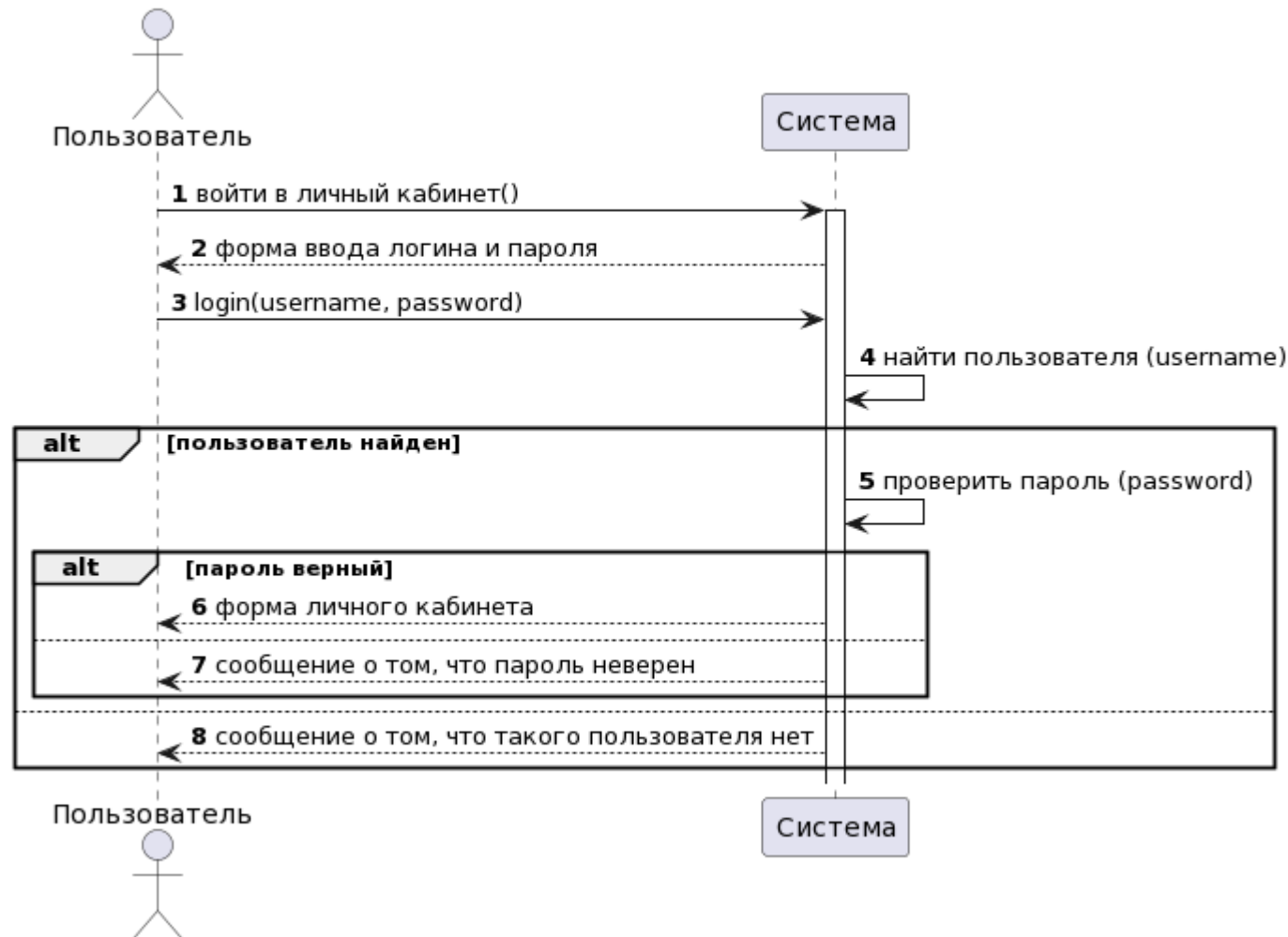
Интернет-магазин: юзкейсы





1. Аутентификация

Аутентификация (вход в систему)

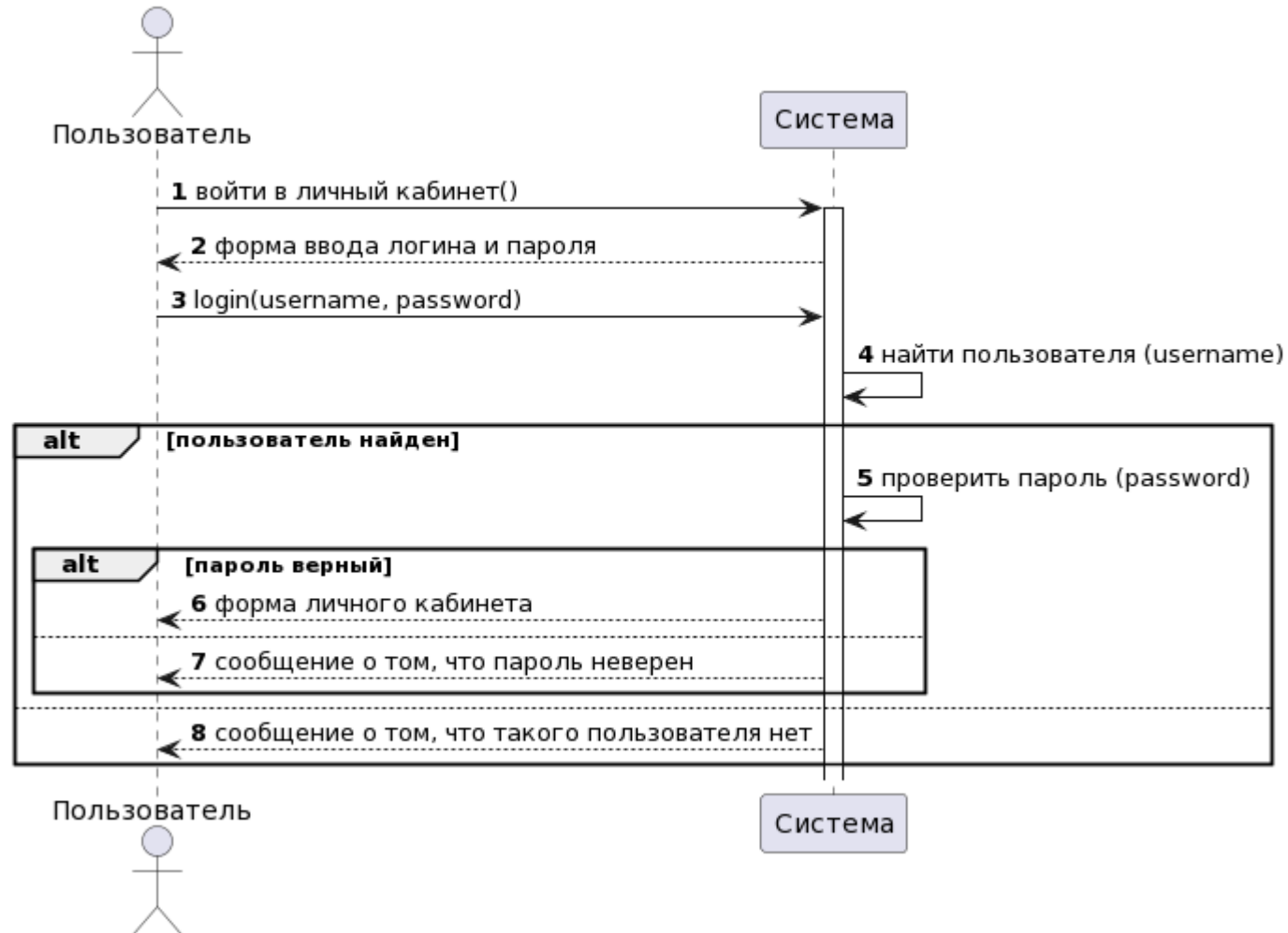


1. Актор Пользователь вызывает у Системы функцию входа в личный кабинет
2. Система возвращает пользователю форму ввода логина и пароля
3. Пользователь вводит логин и пароль и передает их системе с помощью функции `login(username, password)`
4. Система ищет пользователя в своей базе данных по указанному логину
5. Если пользователь найден, система проверяет введенный пароль
6. Если пароль верный, система предоставляет пользователю форму личного кабинета
7. Если пароль неверный, система отправляет сообщение пользователю о том, что пароль неверен
8. Если пользователь не найден, система отправляет сообщение пользователю о том, что такого пользователя нет



1. Аутентификация

Аутентификация (вход в систему)



@startuml

title Аутентификация (вход в систему)

actor Пользователь

participant Система

autonumber

Пользователь -> Система: войти в личный кабинет()

activate Система

Система --> Пользователь: форма ввода логина и пароля

Пользователь -> Система: login(username, password)

Система -> Система: найти пользователя (username)

alt пользователь найден

Система -> Система: проверить пароль (password)

alt пароль верный

Система --> Пользователь: форма личного кабинета

else

Система --> Пользователь: сообщение о том, что пароль

неверен

end alt

else

Система --> Пользователь: сообщение о том, что такого

пользователя нет

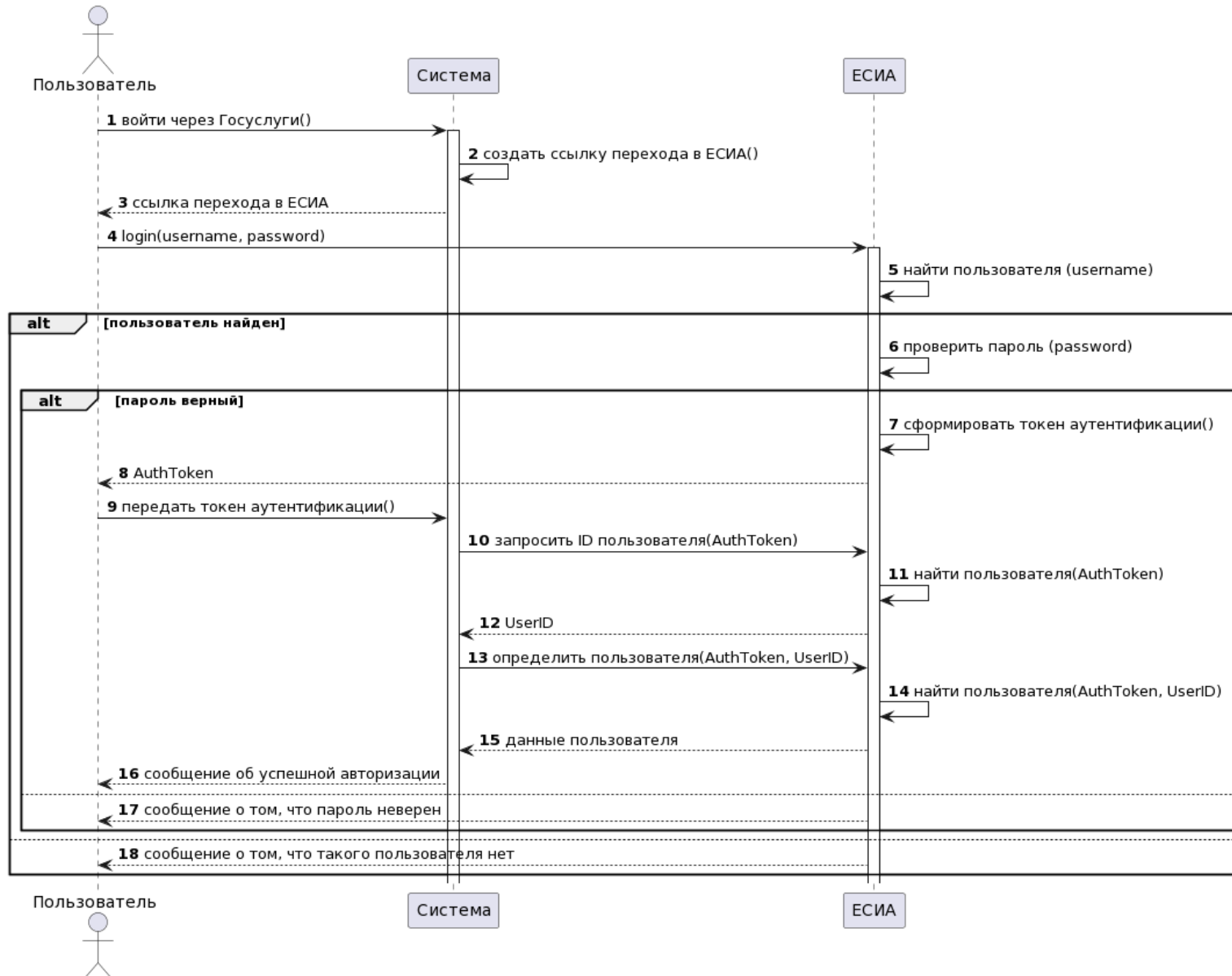
end alt

@enduml

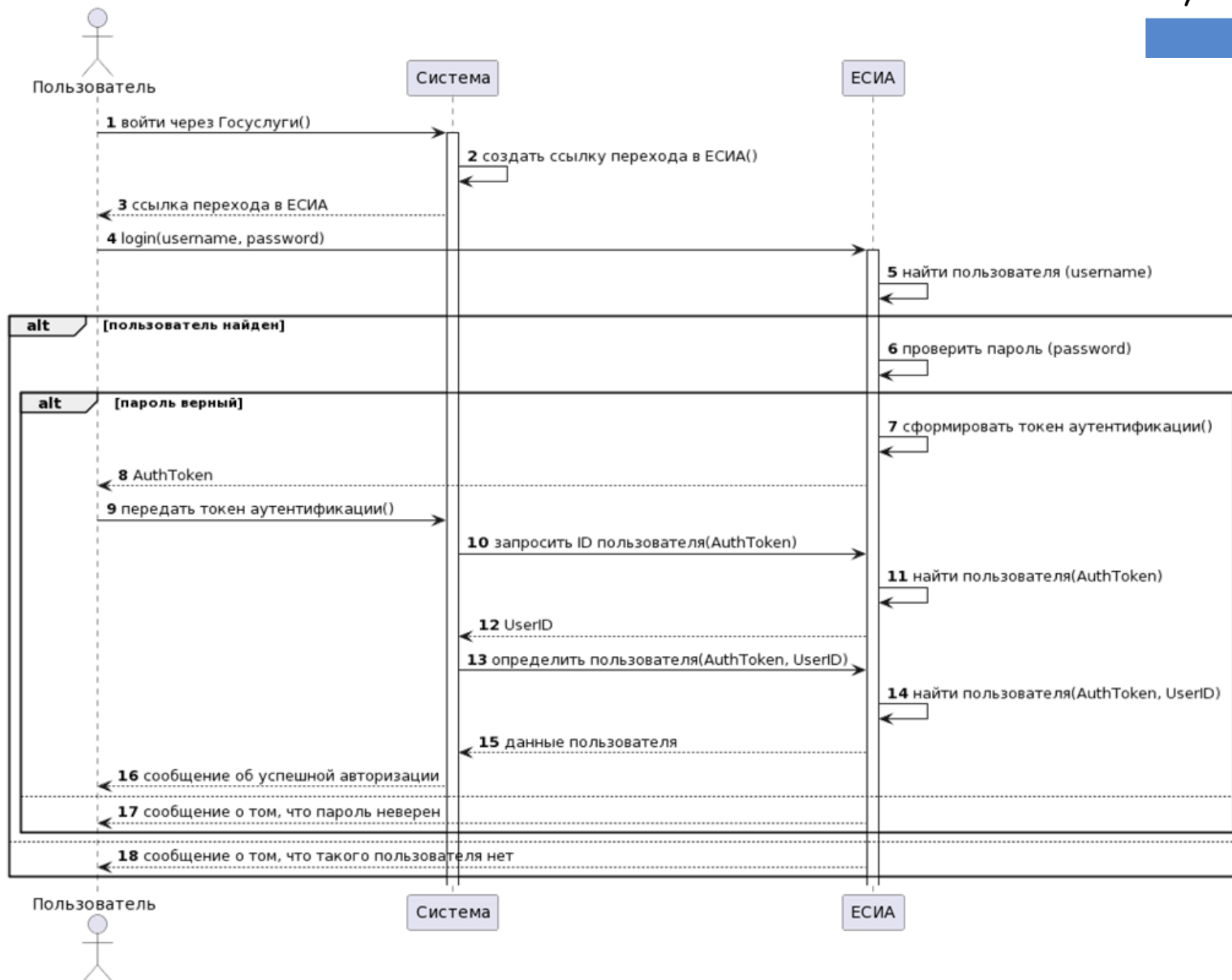


1. Аутентификация через ЕСИА

Аутентификация и авторизация

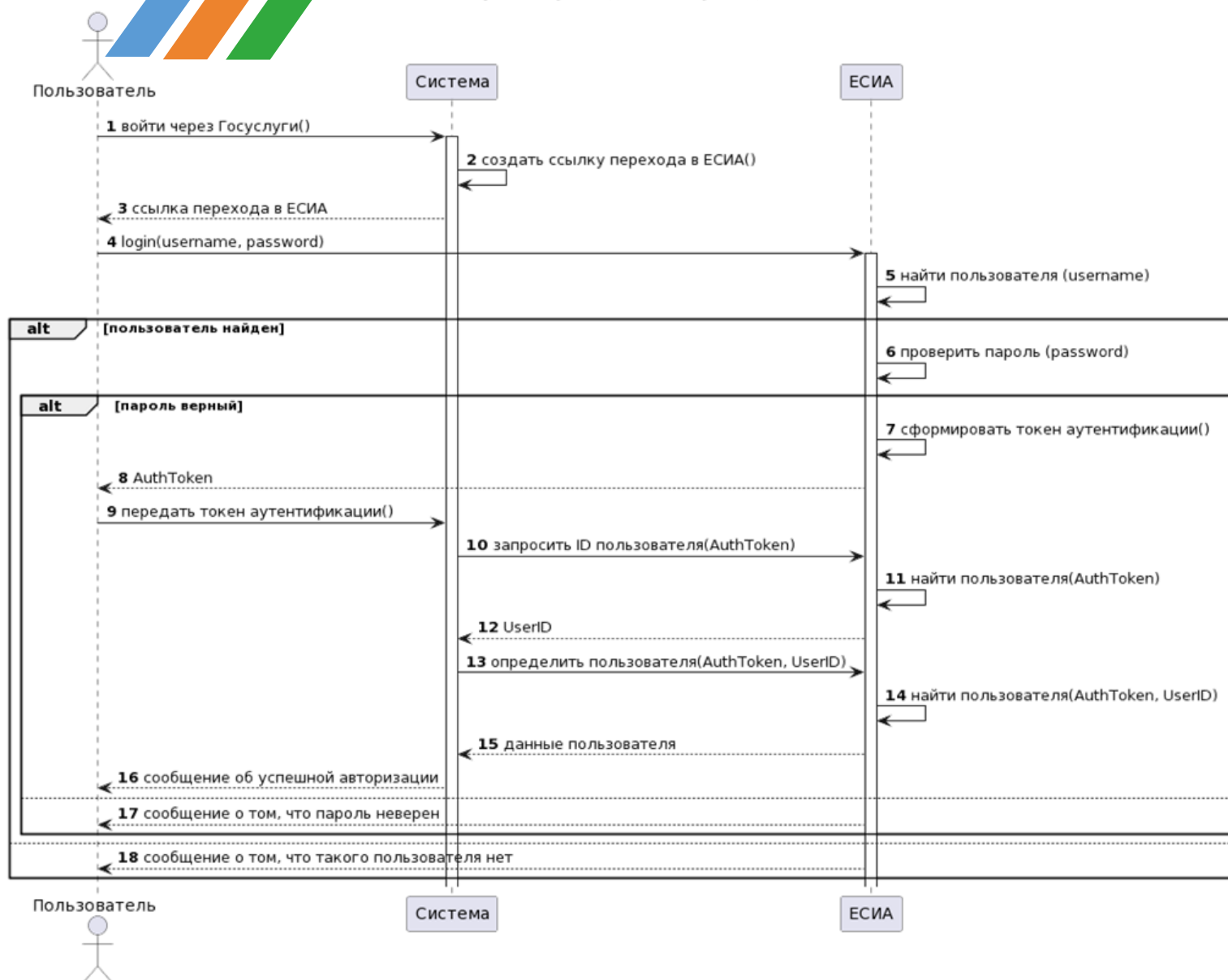


1. Пользователь входит в систему через Госуслуги
2. Система создает ссылку для перехода в ЕСИА
3. Система передает ссылку пользователю
4. Пользователь вводит свое имя пользователя и пароль для входа в ЕСИА
5. ЕСИА ищет пользователя по введенному имени пользователя
6. Если пользователь найден, ЕСИА проверяет введенный пароль
7. Если пароль верный, ЕСИА создает токен аутентификации и передает его пользователю
8. Пользователь передает токен аутентификации системе
9. Система запрашивает у ЕСИА идентификатор пользователя по токenu аутентификации
10. ЕСИА ищет пользователя по токenu аутентификации и передает идентификатор пользователя системе
11. Система определяет пользователя по токenu аутентификации и идентификатору пользователя
12. ЕСИА ищет пользователя по токenu аутентификации и идентификатору пользователя и передает данные пользователя системе
13. Система сообщает пользователю об успешной авторизации
14. Если пароль неверный, ЕСИА сообщает пользователю об ошибке
15. Если пользователь не найден, ЕСИА сообщает пользователю об ошибке.



1. Аутентификация через ЕСИА

Аутентификация и авторизация



@startuml

title Аутентификация и авторизация

actor Пользователь

participant Система

participant ЕСИА

autonumber

Пользователь -> Система: войти через Госуслуги()

activate Система

Система -> Система: создать ссылку перехода в ЕСИА()

Система --> Пользователь: ссылка перехода в ЕСИА

Пользователь -> ЕСИА: login(username, password)

activate ЕСИА

ЕСИА -> ЕСИА: найти пользователя (username)

alt пользователь найден

ЕСИА -> ЕСИА: проверить пароль (password)

alt пароль верный

ЕСИА -> ЕСИА: сформировать токен аутентификации()

ЕСИА --> Пользователь: AuthToken

Пользователь -> Система: передать токен аутентификации()

Система -> ЕСИА: запросить ID пользователя(AuthToken)

ЕСИА -> ЕСИА: найти пользователя(AuthToken)

ЕСИА --> Система: UserID

Система -> ЕСИА: определить пользователя(AuthToken, UserID)

ЕСИА -> ЕСИА: найти пользователя(AuthToken, UserID)

ЕСИА --> Система: данные пользователя

Система --> Пользователь: сообщение об успешной авторизации

else

ЕСИА --> Пользователь: сообщение о том, что пароль неверен

end alt

else

ЕСИА --> Пользователь: сообщение о том, что такого пользователя нет

end alt

@enduml



Практика по кейсам

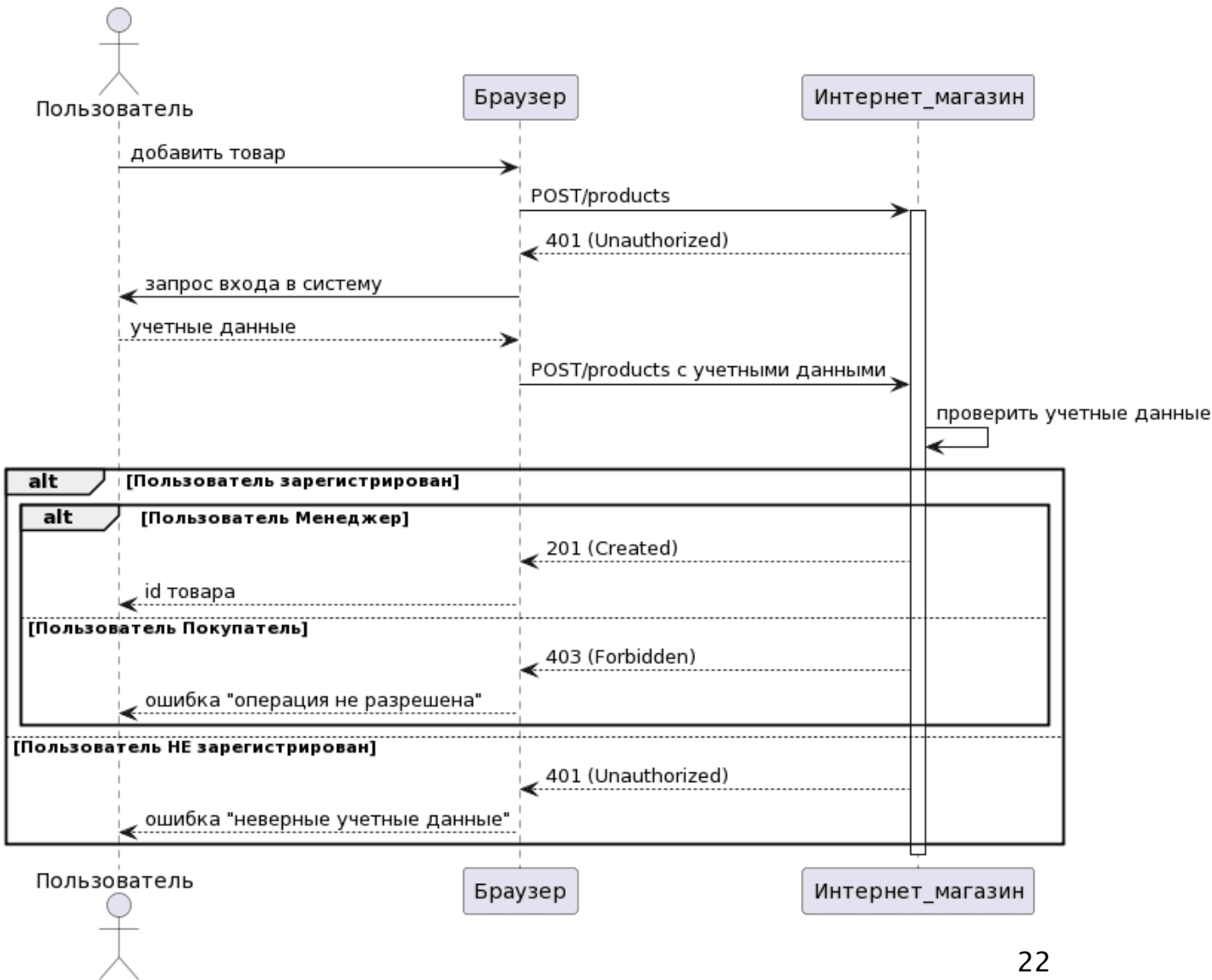
- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.





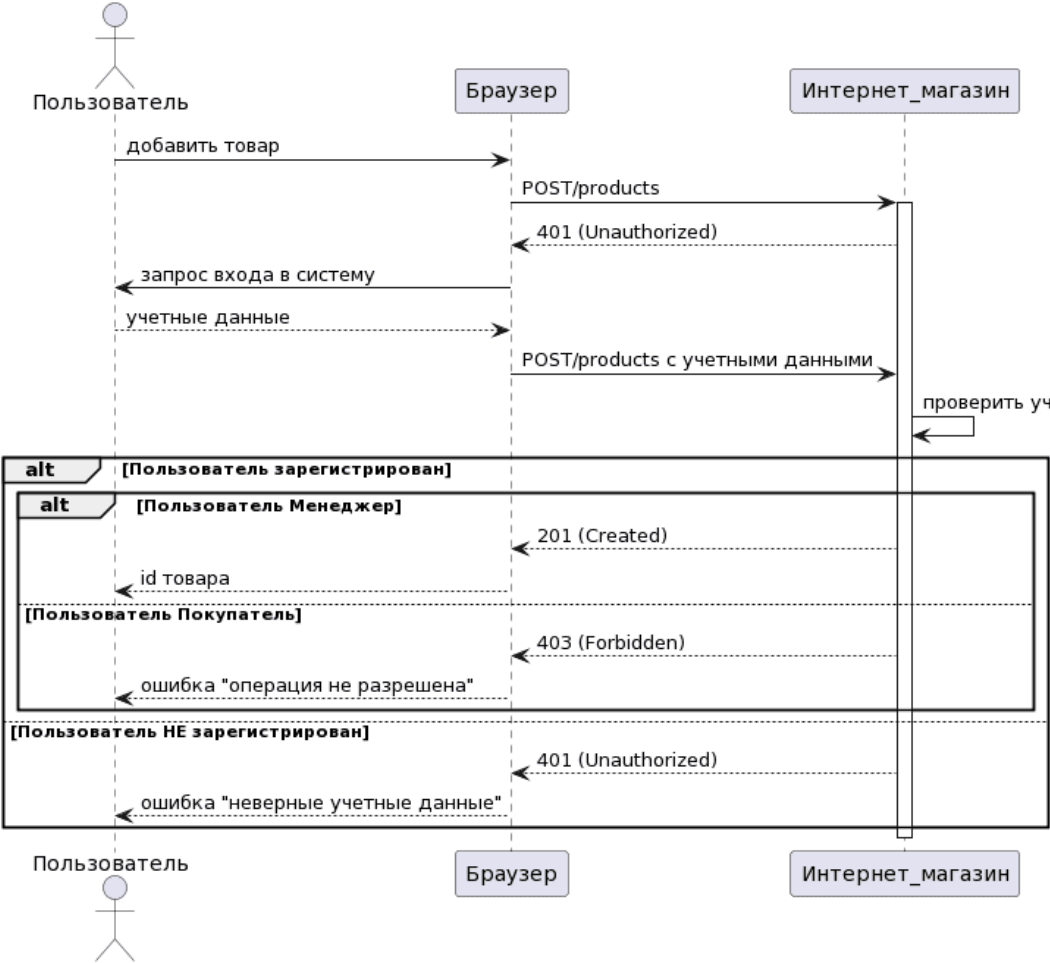
2. Добавить новый товар

1. Пользователь вызывает функцию добавить товар в GUI браузера
2. Браузер отправляет HTTP-запрос POST на добавление товара в интернет-магазин
3. Интернет-магазин активируется и получает запрос. Поскольку пользователь не авторизован, возвращается ответ 401 Unauthorized
4. Браузер запрашивает у пользователя учетные данные для входа в систему
5. Пользователь передает учетные данные браузеру
6. Браузер отправляет магазину HTTP-запрос POST на добавление товара с учетными данными пользователя
7. Интернет-магазин проверяет учетные данные
8. Если пользователь зарегистрирован и является менеджером, кому можно добавлять товары, то интернет-магазин возвращает статус HTTP-ответа 201 Created и браузер передает пользователю идентификатор товара
9. Если пользователь является покупателем, кому нельзя добавлять товары, то интернет-магазин возвращает статус HTTP-ответа 403 Forbidden и браузер передает возвращает ошибку "операция не разрешена".
10. Если пользователь не зарегистрирован, то интернет-магазин возвращает статус HTTP-ответа 401 (Unauthorized) и браузер передает пользователю ошибку "неверные учетные данные".





2. Добавить новый товар



Вход

Форма входа в систему

https:// site-domain-name/product

Имя пользователя

Пароль

Вход Отмена

addproduct.py
http://site-domain-name/product

GUI

Добавить товар

Название: платье

Стоимость: 3520

Бренд: Zolla

Font Doc WS Wasm Manifest Other

requests ☐ 3rd-party requests

☐ Group by frame

☐ Capture screenshots

100 ms 150 ms 200 ms

× Headers Preview Response Initiator >>

▼ General

Request URL: http://site-domain-name/product

Request Method: post

Status Code: 401

Remote Address: xx.xx.xxx.xx:xxx

Referrer Policy: strict-origin-when-cross-origin

▼ Response Headers Заголовки HTTP-ответа

content-type: text/html; charset=iso-8859-1

date: Mon, 20 Feb 2023 04:16:48 GMT

server: nginx

www-authenticate: Basic realm="Thats protected Area!"

▼ Request Headers Заголовки HTTP-запроса

:authority: http://site-domain-name/

:method: post

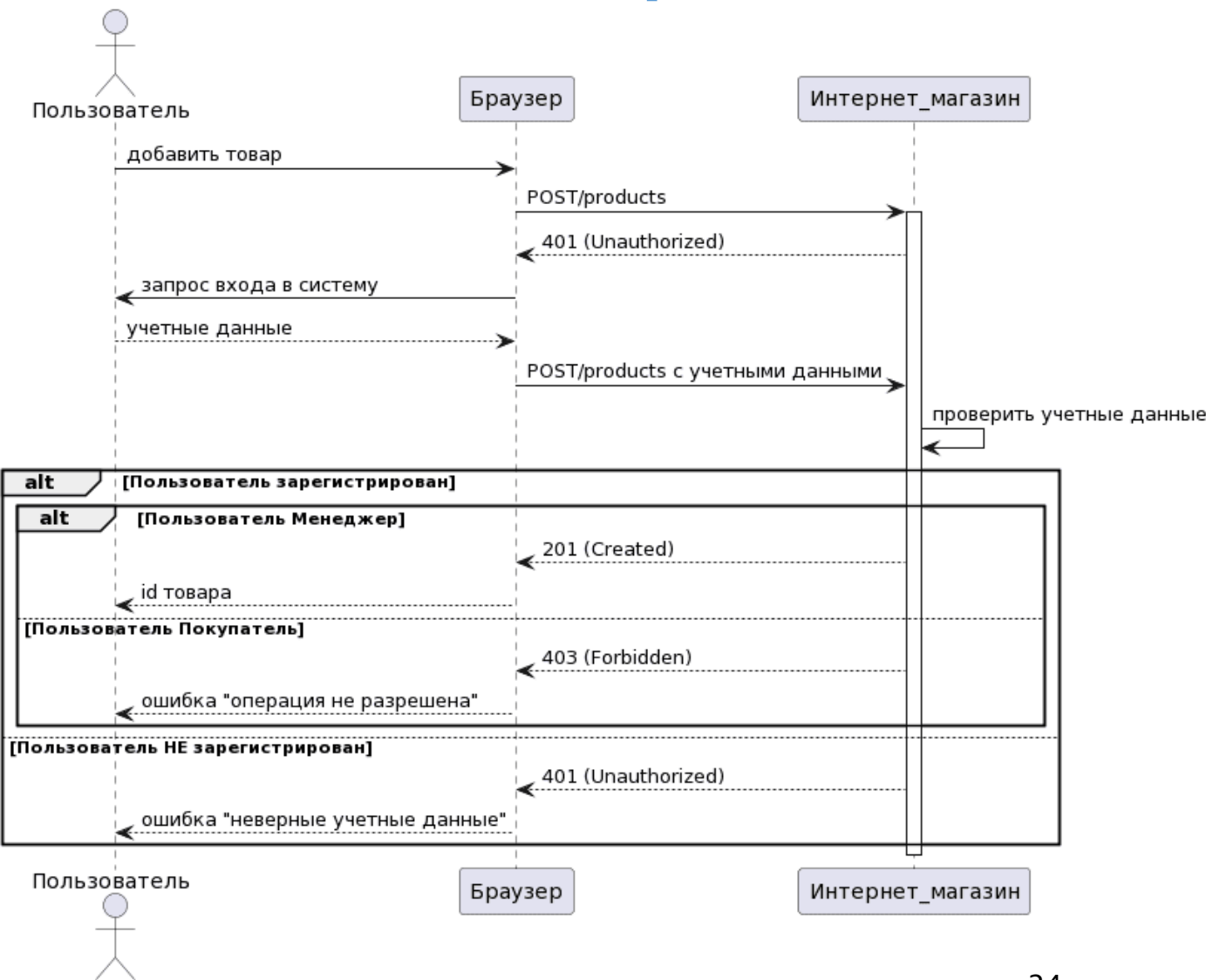
:path: /product/addproduct.py

:scheme: https



2. Добавить новый товар

```
@startuml
actor Пользователь
participant Браузер
participant Интернет_магазин
Пользователь -> Браузер: добавить товар
Браузер -> Интернет_магазин : POST/products
Интернет_магазин --> Браузер : 401 (Unauthorized)
Браузер --> Пользователь : запрос входа в систему
Пользователь --> Браузер : учетные данные
Браузер -> Интернет_магазин : POST/products с учетными данными
Интернет_магазин --> Браузер : 201 (Created)
Браузер --> Пользователь : id товара
else Пользователь Покупатель
Интернет_магазин --> Браузер : 403 (Forbidden)
Браузер --> Пользователь : ошибка "операция не разрешена"
end alt
else Пользователь НЕ зарегистрирован
Интернет_магазин --> Браузер : 401 (Unauthorized)
Браузер --> Пользователь : ошибка "неверные учетные данные"
end alt
deactivate Интернет_магазин
@enduml
```





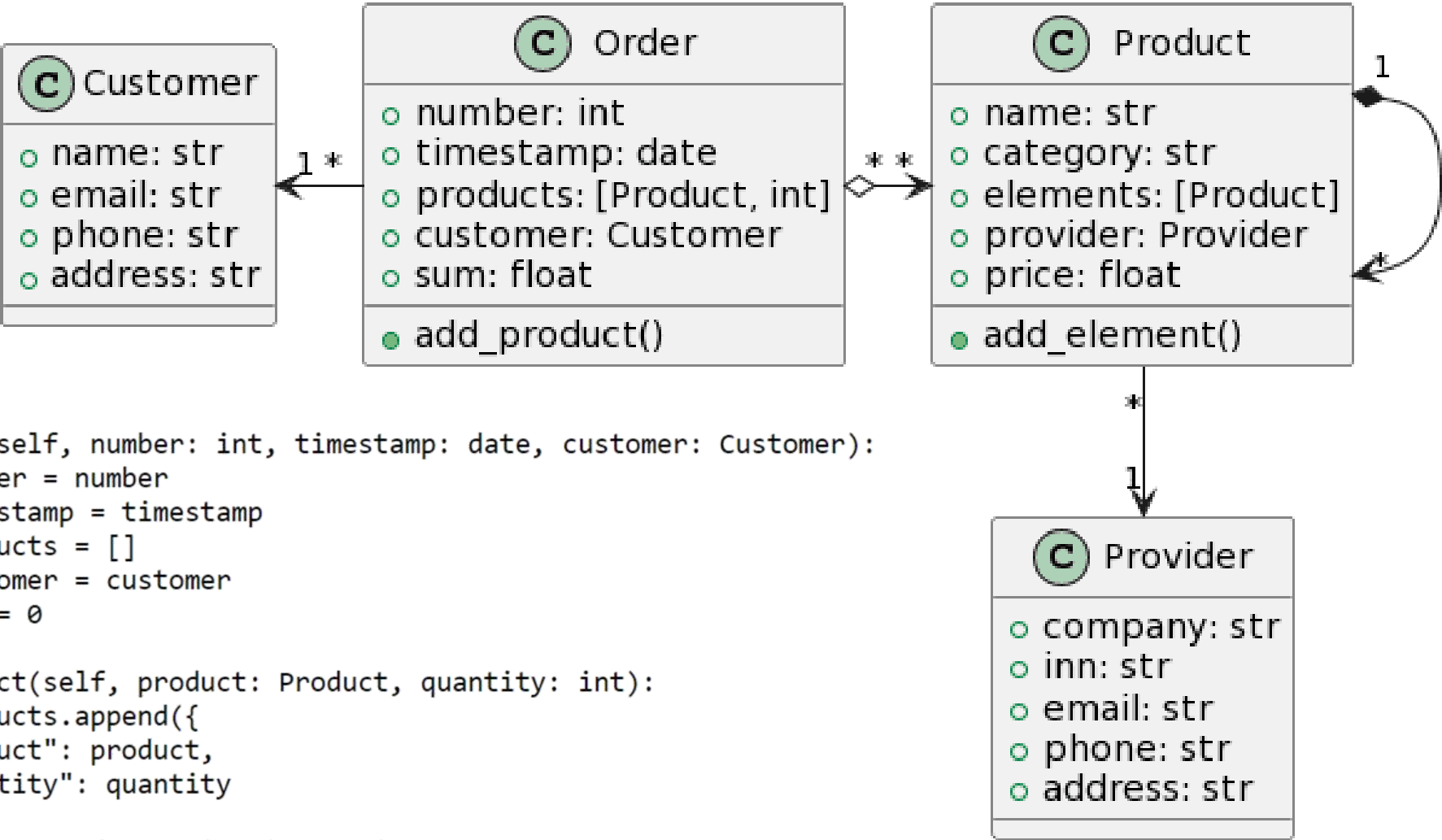
Практика по кейсам

- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.





3. UML-диаграмма классов



```
class Order:
    def __init__(self, number: int, timestamp: date, customer: Customer):
        self.number = number
        self.timestamp = timestamp
        self.products = []
        self.customer = customer
        self.sum = 0

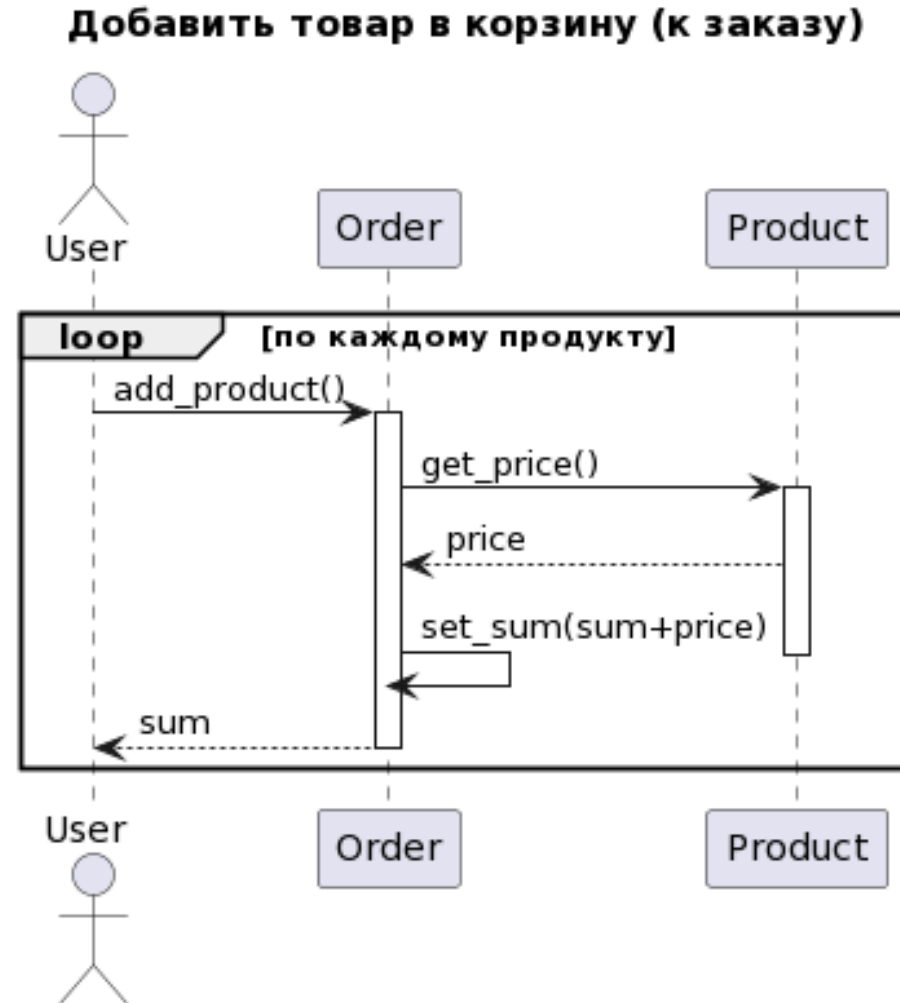
    def add_product(self, product: Product, quantity: int):
        self.products.append({
            "product": product,
            "quantity": quantity
        })
        self.sum += product.price * quantity
```



3. Добавить товар в корзину (к заказу)

Для каждого Товара, добавляемого в Заказ, выполняется следующий цикл:

1. Пользователь вызывает метод `add_product()` у объекта Заказ
2. Заказ запрашивает цену товара, вызывая метод `get_price()` у объекта Товар
3. В ответ объект Товар возвращает информацию о цене
4. Заказ обновляет сумму, увеличивая ее на цену Товара
5. Заказ возвращает пользователю сумму товара



```

class Order:
    def __init__(self, number: int, timestamp: date, customer: Customer):
        self.number = number
        self.timestamp = timestamp
        self.products = []
        self.customer = customer
        self.sum = 0

    def add_product(self, product: Product, quantity: int):
        self.products.append({
            "product": product,
            "quantity": quantity
        })
        self.sum += product.price * quantity
  
```



3. Добавить товар в корзину (к заказу)

@startuml

title Добавить товар в корзину (к заказу)

actor User

participant Order

participant Product

loop по каждому продукту

User -> Order: add_product()

activate Order

Order -> Product: get_price()

activate Product

Product --> Order: price

Order -> Order:

set_sum(sum+price)

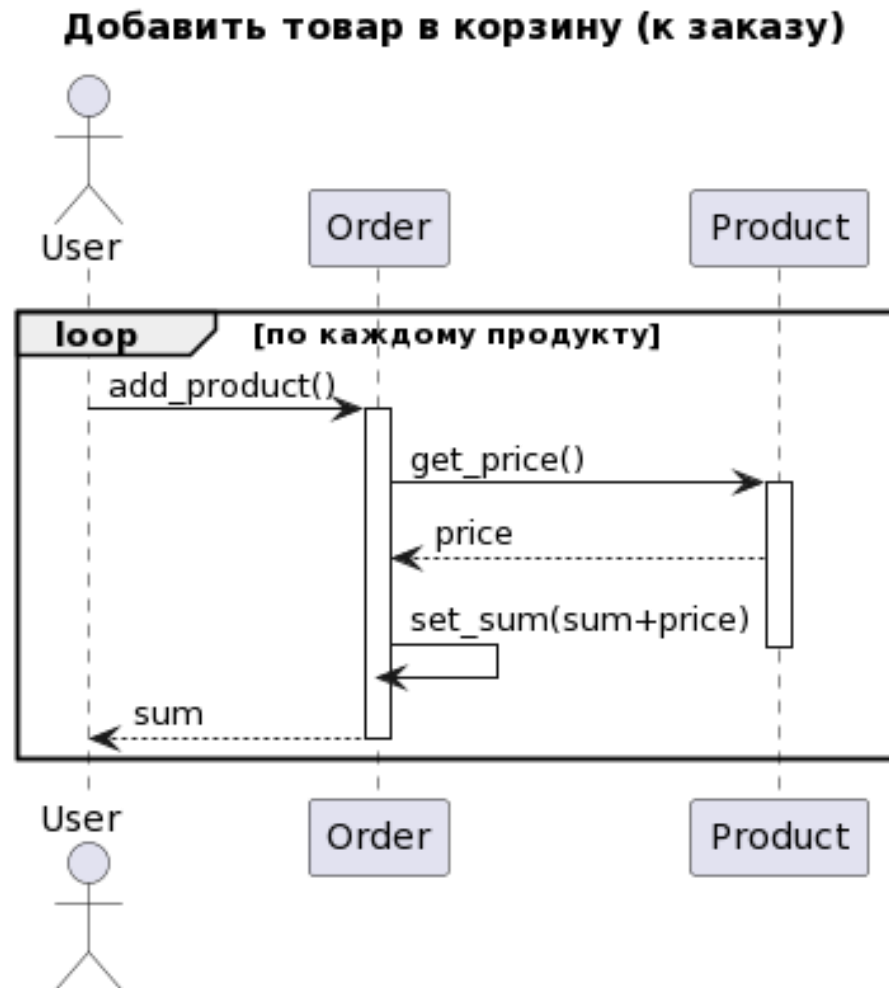
deactivate Product

Order --> User: sum

deactivate Order

end loop

@enduml



```
class Order:
```

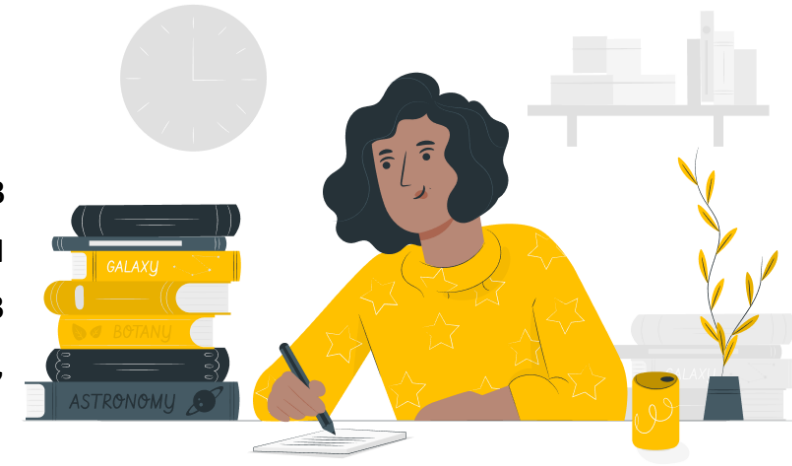
```
def __init__(self, number: int, timestamp: date, customer: Customer):
    self.number = number
    self.timestamp = timestamp
    self.products = []
    self.customer = customer
    self.sum = 0
```

```
def add_product(self, product: Product, quantity: int):
    self.products.append({
        "product": product,
        "quantity": quantity
    })
    self.sum += product.price * quantity
```



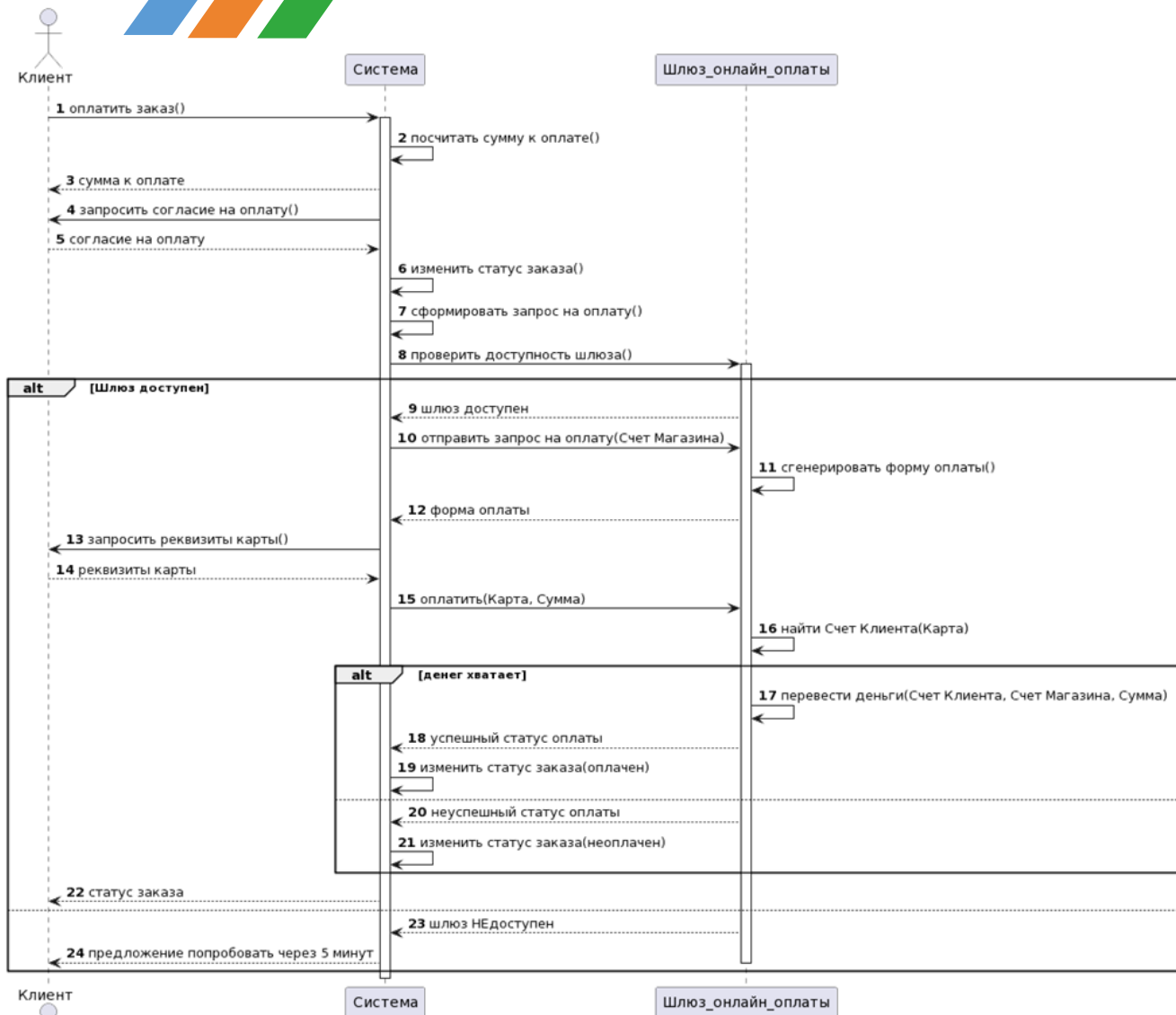
Практика по кейсам

- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.

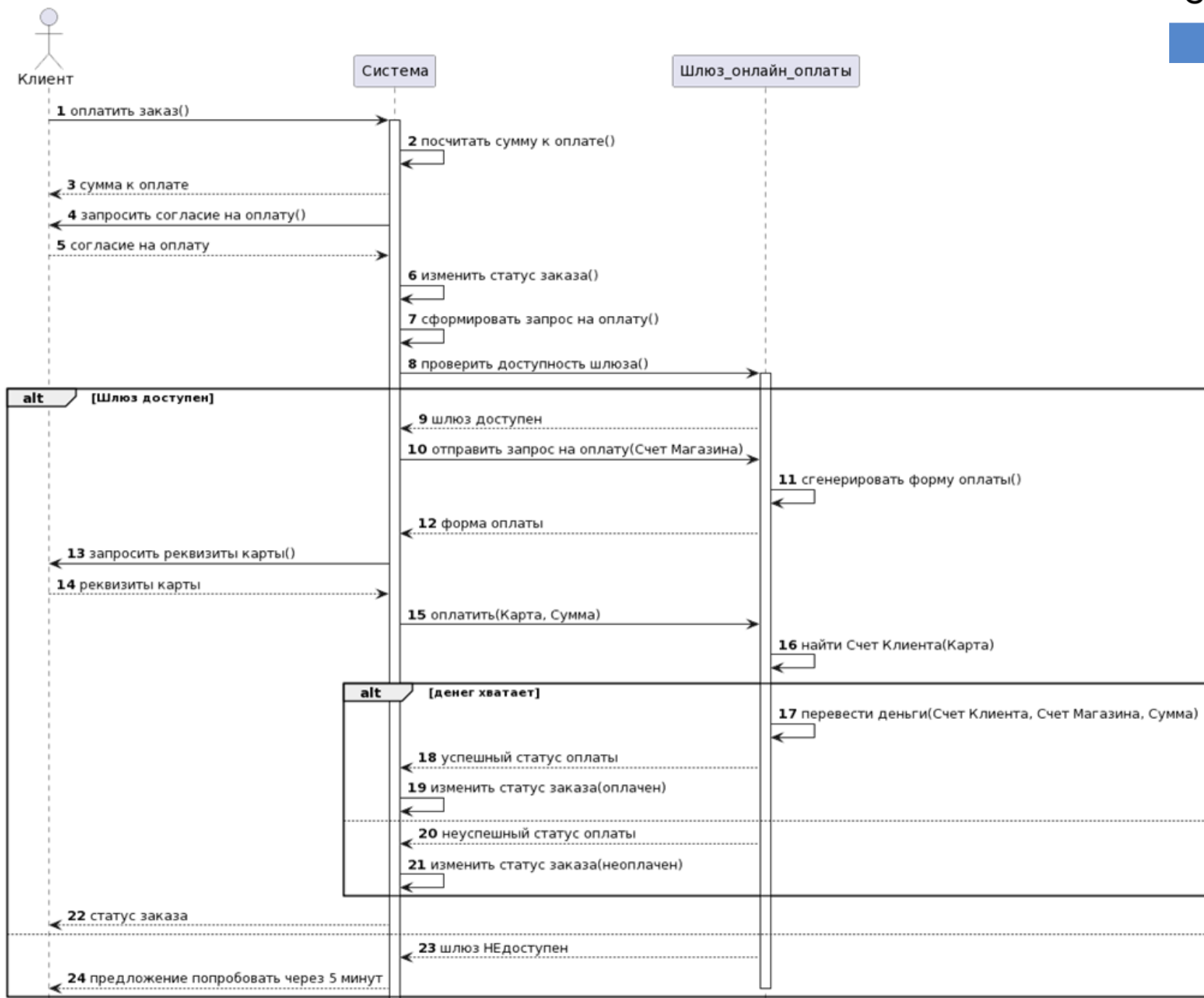


4. Купить товар

Купить товар



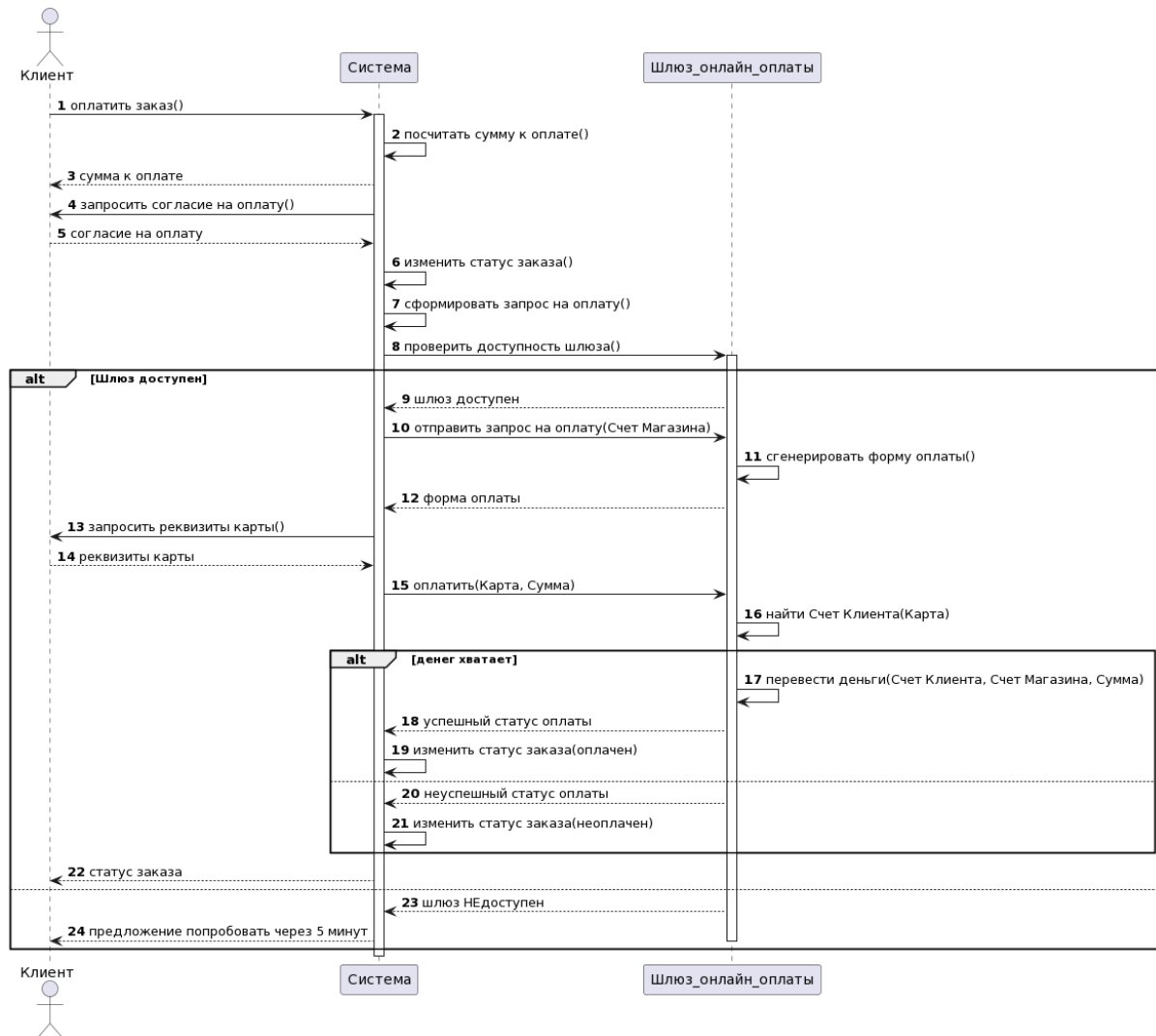
1. Клиент отправляет Системе запрос на оплату заказа
2. Система подсчитывает сумму к оплате и возвращает ее Клиенту
3. Система запрашивает согласие Клиента на оплату
4. Клиент подтверждает согласие на оплату
5. Система изменяет статус заказа и формирует запрос на оплату
6. Система проверяет доступность Шлюза онлайн-оплаты.
7. Если шлюз доступен, Система отправляет запрос на оплату счета магазина Шлюзу онлайн-оплаты
8. Шлюз онлайн-оплаты генерирует форму оплаты и возвращает ее Системе
9. Система запрашивает реквизиты карты у клиента
10. Клиент передает реквизиты карты Системе
11. Система отправляет запрос на оплату с использованием реквизитов карты и суммы
12. Шлюз онлайн-оплаты находит счет клиента
13. Если денег на счету Клиента хватает, Шлюз переводит их со счета Клиента на счет магазина и возвращает успешный статус оплаты Системе
14. Система изменяет статус заказа на "оплачен"
15. Если денег на счету Клиента недостаточно, Шлюз онлайн-оплаты возвращает Системе неуспешный статус оплаты и Система изменяет статус заказа на "неоплачен"
16. Система возвращает Клиенту статус заказа
17. Если Шлюз онлайн-оплаты недоступен, Система предлагает клиенту попробовать оплатить через 5 минут





4. Купить товар

Купить товар



@startuml

title Купить товар

actor Клиент

participant Система

participant Шлюз_онлайн_оплаты

autonumber

Клиент -> Система : оплатить заказ()

activate Система

Система -> Система : посчитать сумму к оплате()

Система --> Клиент : сумма к оплате

Система -> Клиент : запросить согласие на оплату()

Клиент --> Система : согласие на оплату

Система -> Система : изменить статус заказа()

Система -> Система : сформировать запрос на оплату()

Система -> Шлюз_онлайн_оплаты : проверить доступность шлюза()

alt Шлюз доступен

activate Шлюз_онлайн_оплаты

Шлюз_онлайн_оплаты --> Система : шлюз доступен

Система -> Шлюз_онлайн_оплаты : отправить запрос на оплату(Счет Магазина)

Шлюз_онлайн_оплаты -> Шлюз_онлайн_оплаты : сгенерировать форму оплаты()

Шлюз_онлайн_оплаты --> Система : форма оплаты

Система -> Клиент : запросить реквизиты карты()

Клиент --> Система : реквизиты карты

Система -> Шлюз_онлайн_оплаты : оплатить(Карта, Сумма)

Шлюз_онлайн_оплаты -> Шлюз_онлайн_оплаты : найти Счет Клиента(Карта)

alt денег хватает

Шлюз_онлайн_оплаты -> Шлюз_онлайн_оплаты : перевести деньги(Счет Клиента, Счет Магазина, Сумма)

Шлюз_онлайн_оплаты --> Система : успешный статус оплаты

Система -> Система : изменить статус заказа(оплачен)

else

Шлюз_онлайн_оплаты --> Система : неуспешный статус оплаты

Система -> Система : изменить статус заказа(неоплачен)

end alt

Система --> Клиент : статус заказа

else

Шлюз_онлайн_оплаты --> Система : шлюз НЕдоступен

Система --> Клиент : предложение попробовать через 5 минут

deactivate Шлюз_онлайн_оплаты

end alt

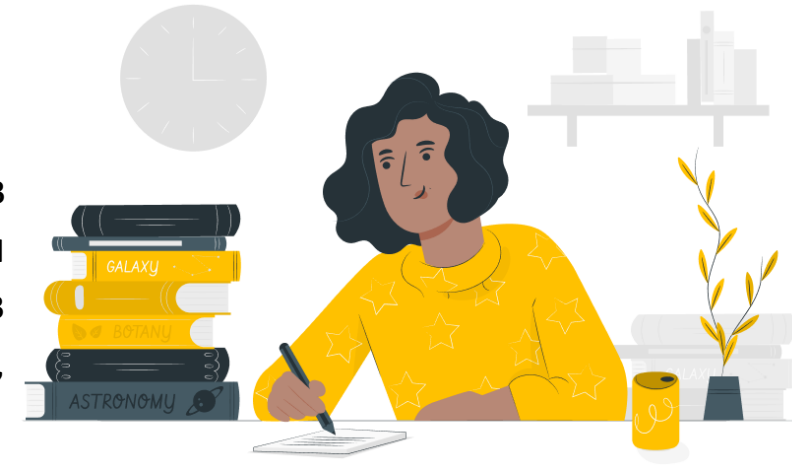
deactivate Система

@enduml



Практика по кейсам

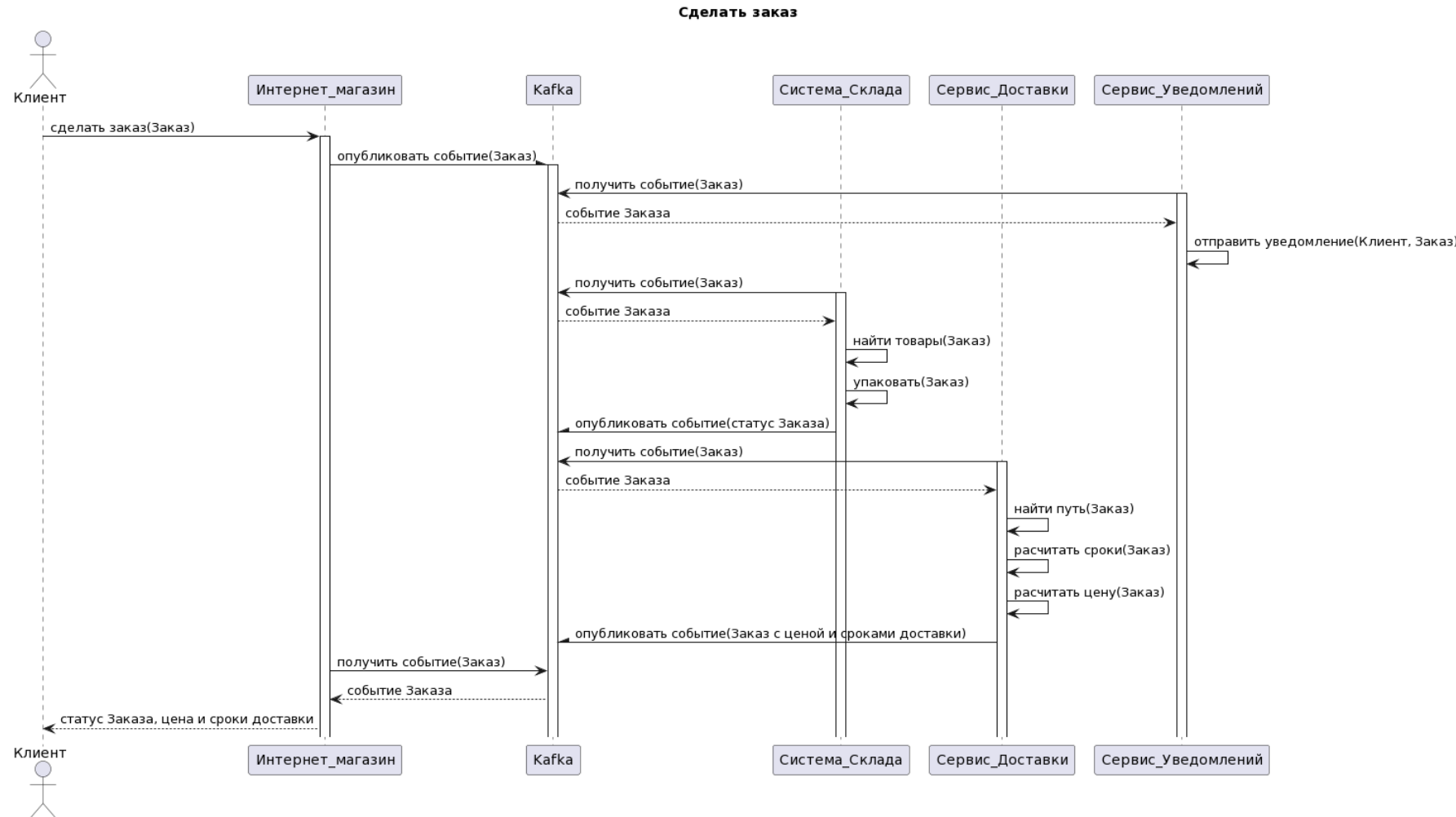
- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.



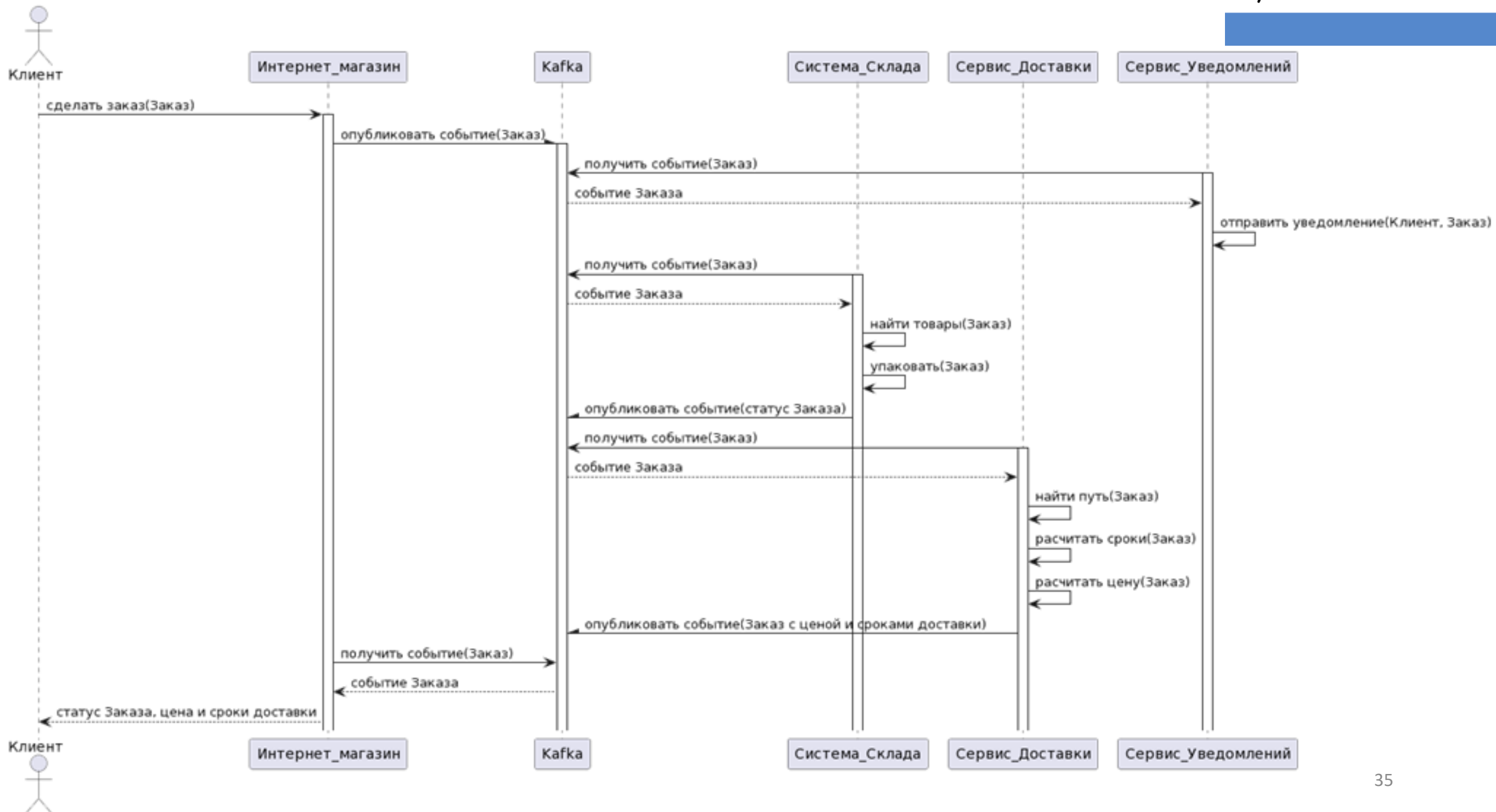


5. Асинхронная интеграция через Kafka

1. Клиент вызывает у Интернет-магазина функцию Сделать заказа
2. Интернет-магазин публикует событие о заказе в Kafka
3. Сервис уведомлений получает событие о заказе из Kafka
4. Сервис Уведомлений отправляет уведомление клиенту
5. Система Склада получает событие о заказе из Kafka и выполняет следующие шаги:
 - находит товары, соответствующие заказу;
 - упаковывает товары;
 - публикует событие о статусе заказа в Kafka
6. Сервис Доставки получает событие о заказе из Kafka и выполняет следующие шаги:
 - находит лучший путь доставки
 - рассчитывает сроки доставки
 - рассчитывает цену доставки.
 - публикует событие о заказе с ценой и сроками доставки в Kafka
7. Интернет-магазин получает событие о заказе из Kafka и передает клиенту информацию о статусе заказа, цене и сроках доставки.



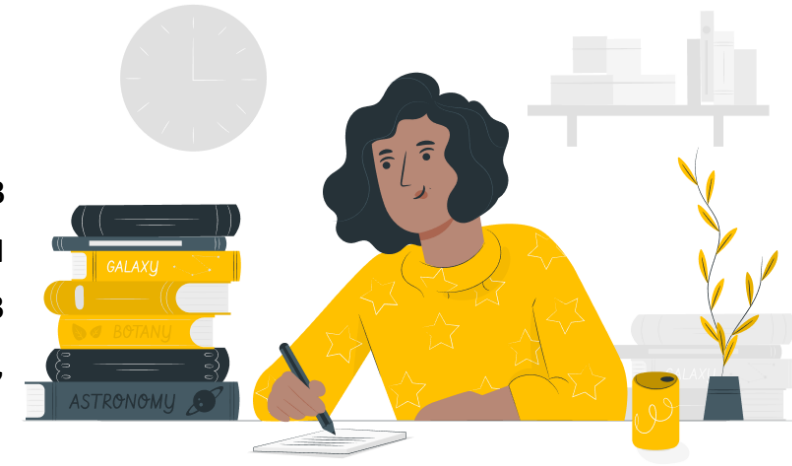
Сделать заказ





Практика по кейсам

- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.

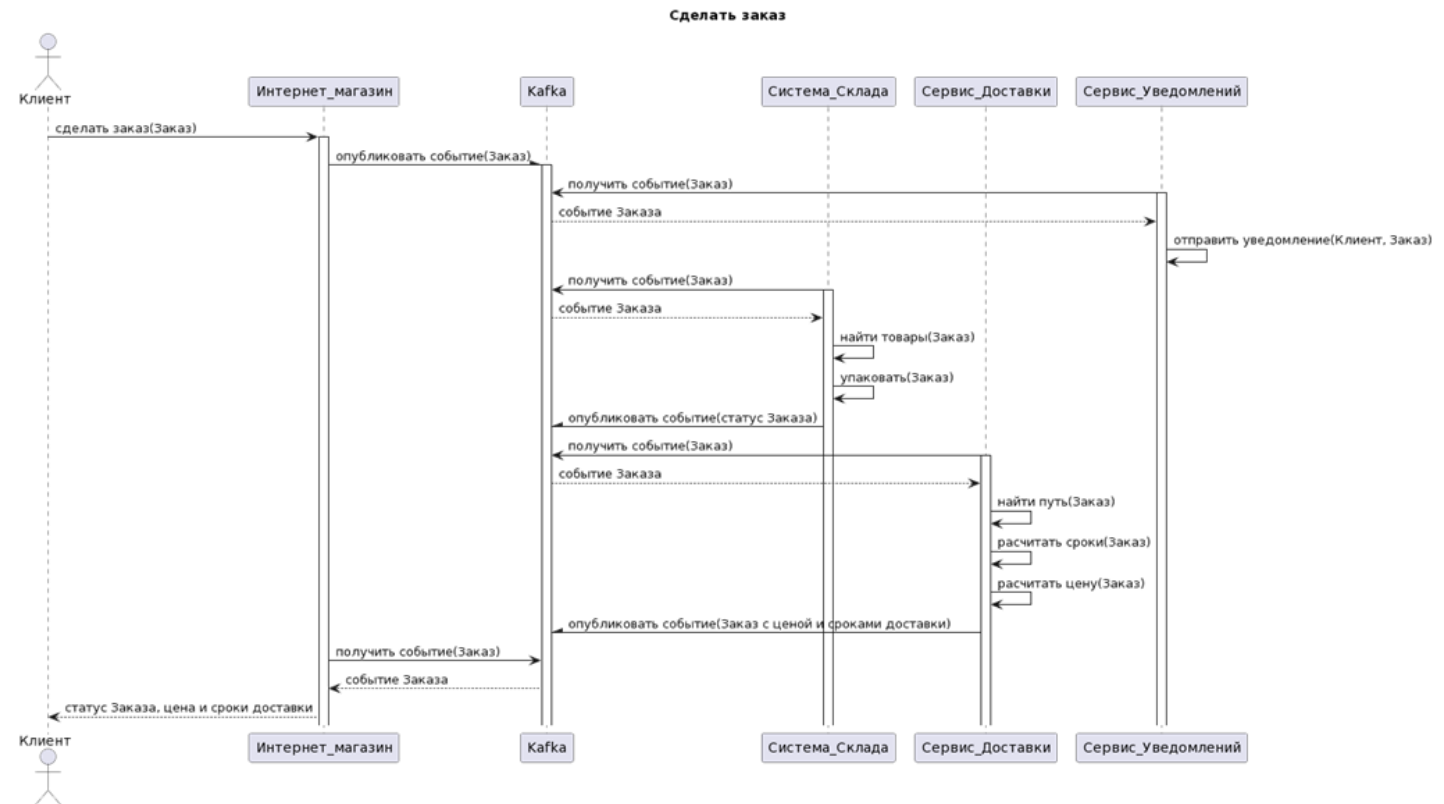




5. Асинхронная интеграция через Kafka

```

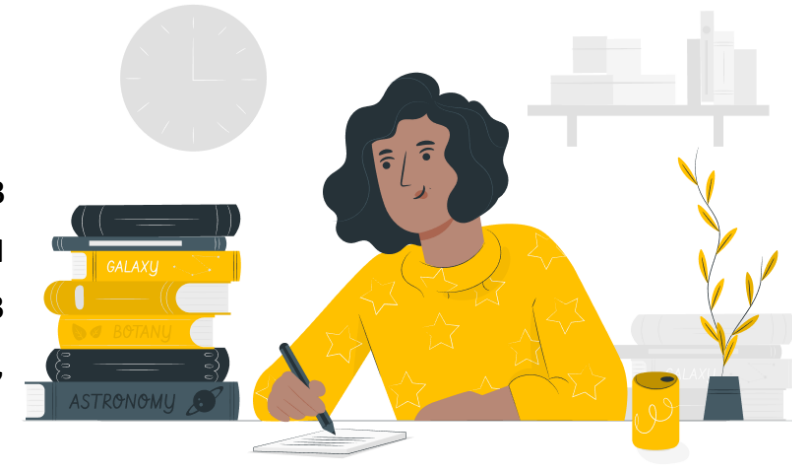
@startuml
title Сделай заказ
actor Клиент
participant Интернет_магазин
participant Kafka
participant Система_Склада
participant Сервис_Доставки
participant Сервис_Уведомлений
Клиент -> Интернет_магазин: сделать заказ(Заказ)
activate Интернет_магазин
Интернет_магазин -\ Kafka: опубликовать событие(Заказ)
activate Kafka
Сервис_Уведомлений -> Kafka: получить событие(Заказ)
activate Сервис_Уведомлений
Сервис_Уведомлений -> Сервис_Уведомлений: отправить уведомление(Клиент, Заказ)
deactivate Сервис_Уведомлений
Kafka --> Система_Склада: получить событие(Заказ)
activate Система_Склада
Система_Склада --> Система_Склада: найти товары(Заказ)
Система_Склада --> Система_Склада: упаковать(Заказ)
Система_Склада -\ Kafka: опубликовать событие(статус Заказа)
deactivate Система_Склада
Kafka --> Система_Склада: получить событие(Заказ)
activate Система_Склада
Система_Склада --> Система_Склада: найти путь(Заказ)
Система_Склада --> Система_Склада: рассчитать сроки(Заказ)
Система_Склада --> Система_Склада: рассчитать цену(Заказ)
Система_Склада -\ Kafka: опубликовать событие(Заказ с ценой и сроками доставки)
deactivate Система_Склада
Kafka --> Сервис_Доставки: получить событие(Заказ)
activate Сервис_Доставки
Сервис_Доставки --> Сервис_Доставки: найти путь(Заказ)
Сервис_Доставки --> Сервис_Доставки: рассчитать сроки(Заказ)
Сервис_Доставки --> Сервис_Доставки: рассчитать цену(Заказ)
Сервис_Доставки -\ Kafka: опубликовать событие(Заказ с ценой и сроками доставки)
deactivate Сервис_Доставки
Интернет_магазин -> Клиент: статус Заказа, цена и сроки доставки
deactivate Интернет_магазин
@enduml
  
```





Практика по кейсам

- **Служба доставки** - фирма обрабатывает заказы на доставку посылок в пределах города за 1 день. Курьер забирает посылку у отправителя и доставляет получателю. Стоимость доставки зависит от веса и габаритов посылки. Для отправителей, которые регулярно пользуются сервисом, предусмотрены скидки согласно программе лояльности.
- **Туроператор** - клиенты бронируют проживание в гостинице на сколько-то суток. Оплата брони может быть онлайн или на месте.
- **Такси** - клиенты заказывают машину такси для поездки из одного места в другое. Стоимость поездки зависит от класса машины такси, наличия у клиента скидки по программе лояльности, времени и длительности поездки.
- **Банк** - Банк выдает клиентам кредиты из линейки готовых кредитных продуктов. Каждый кредитный продукт имеет разный первоначальный взнос, сумму займа, срок кредитования и процентную ставку. После отправки клиентской заявки она проходит автоматический скоринг и при спорных результатах проходит ручную оценку Менеджером банка. Когда рассмотрение заявки окончено, клиенту высылается уведомление об этом по тем каналам связи, которые он оставил в своем профиле.







Узнать подробнее про UML на моих курсах



UML для бизнес-аналитика

<https://babok-school.ru/courses/uml-for-business-analyst/>

- ✓ Бизнес-аналитики, менеджеры и руководители
- ✓ 8 часов
- ✓ Онлайн по будням с 10 до 14 МСК
- ✓ Теория и демо в реальном времени (*без практики онлайн*)
- ✓ Проверка индивидуальной домашней работы на следующий день (опционально, по желанию)
- ✓ 15 тысяч рублей

Основы ООП и разработка UML-моделей

<https://systems.education/uml-mod>

- ✓ Системные аналитики и разработчики
- ✓ 16 часов
- ✓ Онлайн по субботам с 10 до 14 МСК
- ✓ Теория, демо и самостоятельная практика в малых группах онлайн с последующей проверкой преподавателем
- ✓ Проверка вашей работы и обратная связь сразу на занятии
- ✓ 28 тысяч рублей