**Министерство образования и науки Российской Федерации**

**федеральное государственное бюджетное образовательное учреждение высшего профессионального образования**

**«Российский экономический университет имени Г.В. Плеханова»**

Институт цифровой экономики и информационных технологий

Кафедра информатики

**Практическая работа**

по дисциплине

«Объектно-ориентированные технологии в программной инженерии»

Выполнили

студенты 3 курса

группы 15.27Д-БИ19/22Б

очной формы обучения

высшая школа ВШКМиС

Фамилии студентов Нгуен Као Бач,

Москва, 2025

**Задание 1**

A diagram of a company

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

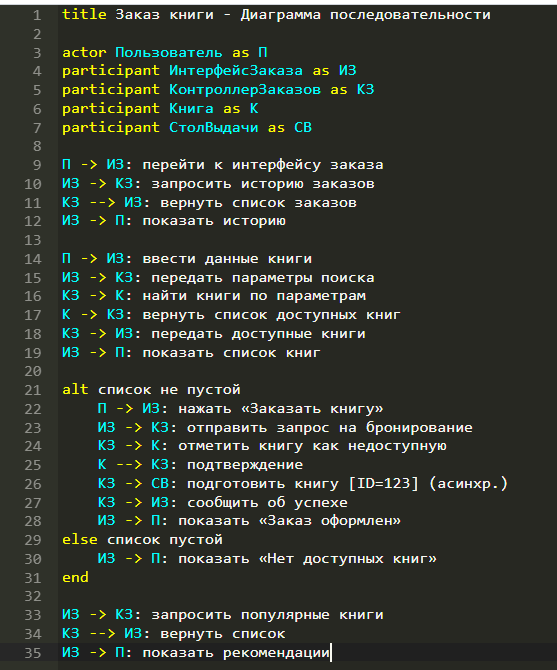
Данная диаграмма последовательности моделирует процесс установления и завершения телефонного звонка между двумя абонентами – A и B. Участвуют следующие объекты: **Абонент A**, **Телефон A**, **Коммутатор**, **Разговор**, **Телефон B** и **Абонент B**.

Процесс делится на 3 этапа:

1. **Инициация звонка**: Абонент A снимает трубку и набирает номер. Телефон A передаёт сигнал и посылает импульсы набора в коммутатор.
2. **Установление соединения**: Коммутатор создаёт объект Разговор и отправляет вызов на Телефон B. Абонент B снимает трубку, соединение устанавливается.
3. **Завершение разговора**: Оба абонента кладут трубку. Разговор завершается и объект Разговор уничтожается.

В диаграмме используются синхронные и асинхронные сообщения, показан момент создания и уничтожения объекта Разговор, а также добавлено пояснение, что абоненты в процессе разговора.

**Задание 2**



A screenshot of a computer screen

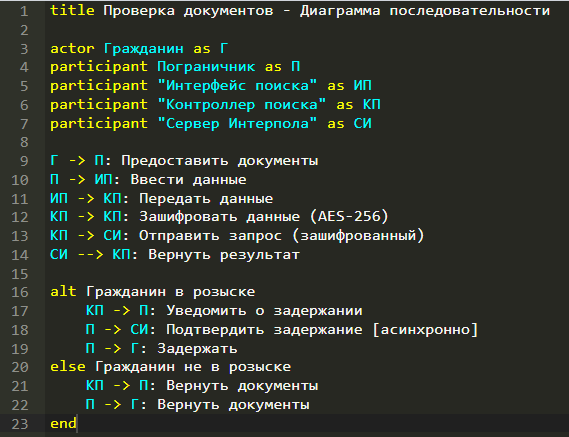
AI-generated content may be incorrect.

Диаграмма включает пять объектов (линий жизни): **Пользователь**, **ИнтерфейсЗаказа**, **КонтроллерЗаказов**, **Книга**, **СтолВыдачи**.

1. Пользователь инициирует взаимодействие, переходя к интерфейсу заказа. Интерфейс запрашивает у контроллера историю предыдущих заказов и отображает её пользователю.
2. Затем пользователь вводит параметры поиска книги. Интерфейс передаёт их контроллеру, который обращается к объекту **Книга** для проверки доступности. Полученные данные возвращаются пользователю.
3. Далее используется конструкция alt для обработки двух сценариев:
   * Если книга найдена (список не пустой), пользователь нажимает кнопку «Заказать книгу». Контроллер отмечает книгу как недоступную и направляет асинхронный запрос на выдачу книги объекту **СтолВыдачи**. После подтверждения пользователю отображается сообщение об успешном оформлении заказа.
   * Если книга недоступна (список пустой), пользователю показывается сообщение об ошибке.
4. В конце пользователь получает рекомендации на основе списка популярных книг, который также запрашивается через контроллер заказов.

Были использованы синхронные и асинхронные сообщения, а также конструкция ветвления alt/else для обработки условий.

**Задание 3**



A diagram with text and arrows

AI-generated content may be incorrect.

Данная диаграмма последовательности моделирует процесс проверки документов гражданина на пограничном контроле через базу данных Интерпола.

В диаграмме участвуют следующие объекты: **Гражданин**, **Пограничник**, **Интерфейс поиска**, **Контроллер поиска** и **Сервер Интерпола**.

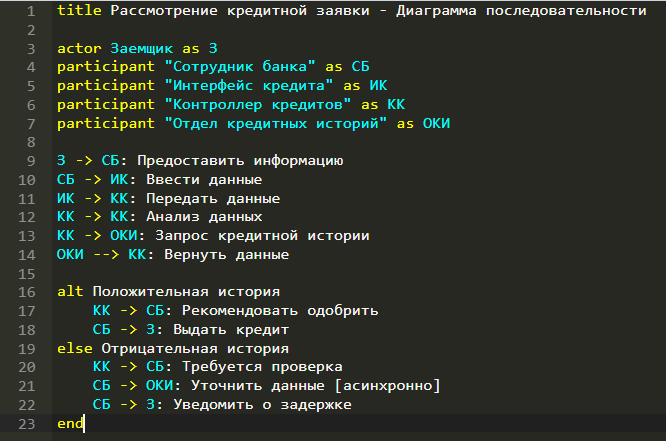
Сценарий начинается с передачи документов пограничнику. Далее данные вводятся в систему через интерфейс и передаются контроллеру, где шифруются по алгоритму **AES-256**, затем отправляются на сервер Интерпола.

После получения результата реализуется ветвление:

* **Если гражданин в розыске**, пограничник получает уведомление, подтверждает задержание (асинхронно) и задерживает гражданина.
* **Если гражданин не в розыске**, документы возвращаются обратно.

Диаграмма показывает реалистичный и корректный процесс обработки, включает ветвление alt/else, различие между синхронными и асинхронными сообщениями.

**Задание 4**



A diagram of a company

AI-generated content may be incorrect.

Данная диаграмма последовательности отражает процесс рассмотрения кредитной заявки клиента в банке.

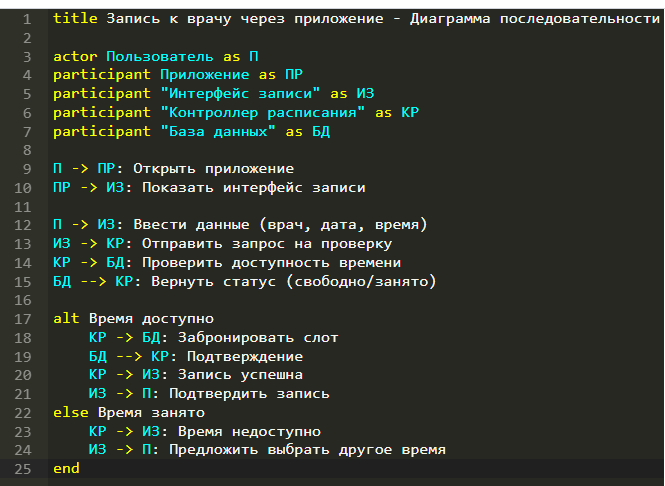
Участвуют следующие объекты: **Заемщик**, **Сотрудник банка**, **Интерфейс кредита**, **Контроллер кредитов**, и **Отдел кредитных историй**.

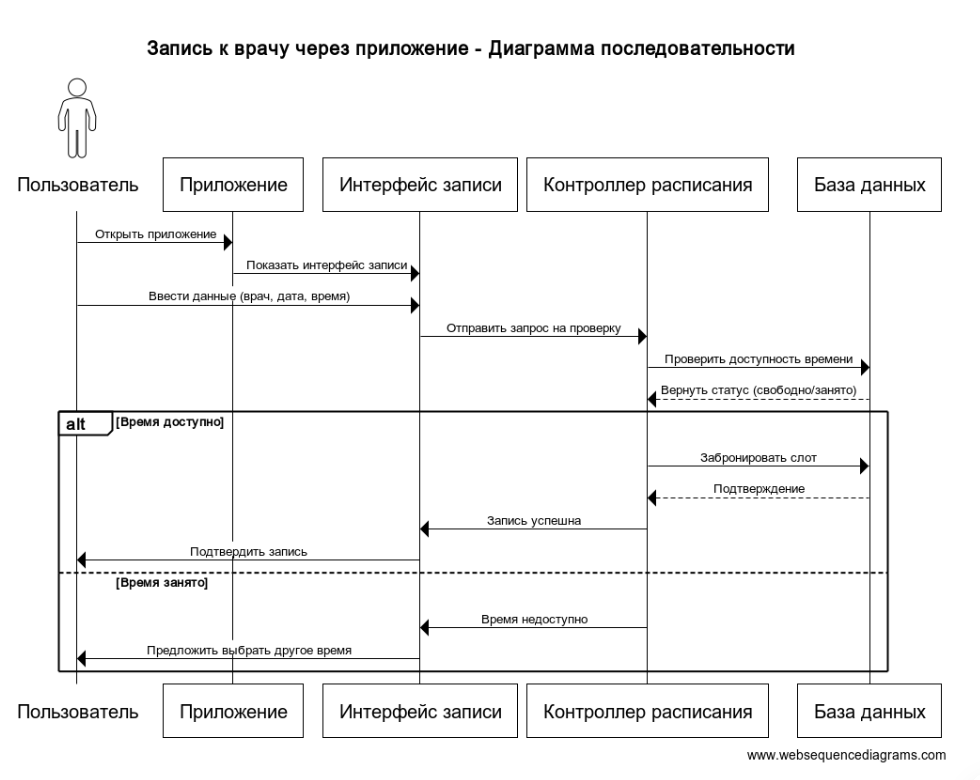
Процесс начинается с предоставления заемщиком информации и цели кредита. Сотрудник банка вводит данные в систему через интерфейс. Контроллер анализирует полученные данные и направляет запрос в отдел кредитных историй.

В зависимости от результата:

* При **положительной истории** контроллер рекомендует одобрить заявку, и кредит выдается.
* При **отрицательной истории** сотрудник запрашивает дополнительную информацию в отделе (асинхронно) и уведомляет заемщика о задержке рассмотрения заявки.

**Задание 5**





Данная диаграмма последовательности моделирует процесс записи к врачу через мобильное приложение.

В диаграмме участвуют следующие объекты: Пользователь, Приложение, Интерфейс записи, Контроллер расписания, и База данных.

Процесс начинается с того, что пользователь открывает приложение и переходит в интерфейс записи. Затем он вводит данные — врача, дату и время. Интерфейс передаёт запрос контроллеру расписания, который обращается к базе данных для проверки доступности выбранного времени.

Если время доступно, слот бронируется, и пользователю отображается подтверждение записи. Если время занято, система информирует пользователя и предлагает выбрать другое время.

Контрольные вопросы

**1. Перечислите особенности диаграммы последовательности.**  
Диаграмма последовательности — это инструмент UML, который отображает взаимодействие объектов в хронологическом порядке. Её особенности включают:

* Фокус на временной последовательности сообщений между объектами.
* Использование линий жизни (lifeline) для отображения периода активности объектов.
* Возможность отображать создание и уничтожение объектов, ветвления и асинхронные взаимодействия.

**2. С какой целью создаются диаграммы последовательности?**

Эти диаграммы создаются для:

* Визуализации логики выполнения сценариев в системе.
* Анализа взаимодействия между объектами или компонентами.
* Документирования поведения системы для разработчиков и заинтересованных лиц.

**3. Перечислите элементы нотации диаграммы последовательности.**

Основные элементы:

* **Объекты** (прямоугольники с именами и классами).
* **Линии жизни** (вертикальные пунктирные линии).
* **Сообщения** (стрелки с подписями, например, синхронные или асинхронные).
* **Фокус управления** (узкие прямоугольники на линиях жизни, показывающие активность).
* **Условные блоки** (ветвления с квадратными скобками).

**4. Что такое объект?**

Объект — это экземпляр класса, участвующий во взаимодействии. На диаграмме он обозначается прямоугольником с именем и классом (например, Абонент: Клиент).

**5. Что такое линия жизни?**

Линия жизни — это вертикальная пунктирная линия, идущая от объекта. Она показывает период его существования и активности в системе. Если объект уничтожается, линия перечёркивается крестом.

**6. Что такое сообщение?**

Сообщение — это передача информации или запроса между объектами. Изображается стрелкой между линиями жизни. Пример: вызов метода (поднятьТрубку()) или асинхронный сигнал.

**7. Какие виды сообщений используются на диаграмме последовательности? Приведите примеры.**

* **Синхронные** (сплошная стрелка с треугольником): вызов метода с ожиданием ответа (например, запроситьДанные()).
* **Асинхронные** (стрелка с открытым наконечником): сигнал без ожидания (например, отправитьУведомление()).
* **Возвратные** (пунктирная стрелка): ответ на синхронный запрос (например, возврат результата вычислений).
* **Создающие** (стрелка с подписью create): инстанциирование нового объекта.

Примеры из лабораторной работы:

* Асинхронное сообщение набратьНомер() между абонентом и телефонным аппаратом.
* Синхронный запрос от коммутатора к объекту Разговор.