ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 5
СПЕЦИФИКАЦИЯ ВЗАИМОДЕЙСТВИЯ ПОДСИСТЕМ

Дисциплина: CSE842 «Capstone Project». **Образовательная программа:** 6B06102 «Computer Science», 4 курс, 8 семестр. **Связь с лекцией:** лекция 5 «Проектирование данных и интерфейсов подсистем». **Результаты обучения:** РО4. **Объём:** 3 часа практики, 7 часов самостоятельной работы, включая СРСП. **Максимальный балл:** 3.

Цель занятия: подготовить модель данных и контракт интерфейса подсистем с примерами успешного и ошибочного обмена.

Задачи:

1. Построить модель предметной области и ER-диаграмму, определить инварианты.
2. Разработать физическую модель данных (SQL-скрипт) с ограничениями и индексами.
3. Описать контракт ключевых операций API (OpenAPI или таблица) с кодами ошибок.
4. Описать контракт асинхронного события.
5. Подготовить примеры успешного и ошибочного обмена.

Программное обеспечение и инструменты: СУБД SQLite или PostgreSQL, Swagger Editor, Python 3 (проверка примеров), Mermaid erDiagram.

1. Краткие теоретические сведения

Инварианты предметной области обеспечиваются на трёх уровнях: ограничения БД (NOT NULL, CHECK, FK, UNIQUE, EXCLUDE), проверки ядра внутри транзакции и проверка формата на интерфейсе. Контракт операции содержит вход, предусловия, постусловия, ошибки и нефункциональные гарантии. Порядок проверок (400 → 401 → 422 → 404 → 409) — часть контракта.

Пример ограничения и индекса:

```
CREATE TABLE booking (
  id INTEGER PRIMARY KEY,
  equipment_id INTEGER NOT NULL
    REFERENCES equipment(id),
  start_ts TEXT NOT NULL,
  end_ts TEXT NOT NULL,
  CHECK (start_ts < end_ts));
CREATE INDEX ix_booking_eq_time
  ON booking(equipment_id, start_ts);
```

Таблица 1 — Пример фрагмента контракта

Запрос	Код	Ответ
корректная бронь	201	{"id": 17, "status": "confirmed"} + Location
интервал занят	409	{"code": "SLOT_TAKEN", "next_free": "…"}

Запрос	Код	Ответ
equipment_id = "3"	422	{"code": "FIELD", "field": "equipment_id"}
повреждённый JSON	400	{"code": "INVALID_JSON"}

2. Порядок выполнения работы

1. Постройте модель предметной области по сущностям варианта (таблица 2); выпишите инварианты.
2. Постройте ER-диаграмму и нормализуйте модель до 3НФ.
3. Напишите SQL-скрипт создания схемы с ограничениями и индексами для частых запросов; выполните его.
4. Опишите в OpenAPI ключевую операцию варианта и 3–5 других операций; укажите схемы и коды ответов.
5. Подготовьте не менее 6 примеров обмена (успех и ошибки) в каталоге contracts/examples и проверьте их скриптом.
6. Опишите контракт события (имя, версия, поля, гарантия доставки, идемпотентность получателя).
7. Согласуйте контракт между участниками, отвечающими за разные подсистемы.

3. Индивидуальные задания

Номер варианта совпадает с номером темы проекта, выбранной на практическом занятии № 1, и сохраняется до конца семестра. Команда (до 4 человек) выполняет задание своего варианта; распределение работы между участниками указывается в отчёте.

Таблица 2 — Индивидуальные задания: данные и интерфейс

№	Сущности	Ключевая операция	Обязательные ошибочные ответы
1	Пациент, Врач, Слот, Запись	POST /appointments	409 слот занят; 422 слот в прошлом; 404 врач не найден
2	Студент, Занятие, Токен, Отметка	POST /attendance	410 токен истёк; 409 уже отмечен; 403 не в группе
3	Датчик, Измерение, Порог, Оповещение	GET /sensors/{id}/measurements?from&to	400 неверный интервал; 404 датчик; 413 слишком большой период
4	Задание, Тест, Решение, Результат	POST /submissions	413 файл велик; 415 язык не поддержан; 429 лимит отправок
5	Товар, Остаток, Корзина, Заказ	POST /orders	409 нет остатка; 402 оплата отклонена; 422 пустая корзина

№	Сущности	Ключевая операция	Обязательные ошибочные ответы
6	Книга, Экземпляр, Читатель, Выдача	POST /loans	409 экземпляр выдан; 403 долг читателя; 404 штрихкод
7	Вопрос, Статья, Обращение, Оценка	POST /questions	429 лимит сообщений; 503 поиск недоступен; 400 пустой вопрос
8	Парковка, Место, Наблюдение, Камера	GET /lots/{id}/spots	404 парковка; 503 нет данных камеры; 200 с меткой устаревания
9	Товар, Ячейка, Операция, Остаток	POST /operations	409 недостаточно товара; 404 штрихкод; 422 количество ≤ 0
10	Заказ, Курьер, Позиция, Ресторан	POST /orders/{id}/assign	409 заказ уже назначен; 404 курьер; 422 курьер не на смене
11	Запись журнала, Шаблон, Окно, Оповещение	GET /alerts?since	400 неверный параметр; 401; 503 детектор недоступен
12	Счёт, Операция, Категория, Чек	POST /receipts	422 сумма не найдена; 413 большое фото; 409 повторный чек
13	Заявление, Шаблон, Согласование, Документ	POST /applications/{id}/approve	403 нет права; 409 уже согласовано; 423 документ подписан
14	Площадка, Слот, Блокировка, Платёж	POST /slots/{id}/hold	409 слот заблокирован; 410 блокировка истекла; 402 оплата
15	Устройство, Событие, Правило, Команда	POST /rules	422 цикл правил; 404 устройство; 409 конфликт правил
16	Пропуск, Транспорт, Проезд, Камера	POST /passes/check	403 пропуск недействителен; 422 номер не распознан; 404 камера
17	Счётчик, Показание, Прогноз, Оповещение	GET /meters/{id}/forecast	404 счётчик; 409 мало данных; 400 горизонт

№	Сущности	Ключевая операция	Обязательные ошибочные ответы
18	Тест, Вопрос, Попытка, Ответ	PUT /attempts/{id}/answers/{q}	409 время вышло; 403 чужая попытка; 422 неверный формат
19	Остановка, Рейс, Пересадка, Маршрут	GET /journeys? from&to&time	404 остановка; 422 время в прошлом; 204 маршрут не найден
20	Преподаватель, Слот, Запись, Внешнее событие	POST /consultations	409 пересечение; 401 нет доступа к календарю; 404 слот

4. Содержание отчёта

1. Модель предметной области, ER-диаграмма, инварианты.
2. SQL-скрипт схемы и результат его выполнения.
3. Спецификация OpenAPI (или таблица операций).
4. Примеры обмена и результат их проверки.
5. Контракт события.

5. Форма сдачи

Артефакты занятия размещаются в репозитории проекта (каталог docs/ или указанный в задании), в трекере задач создаются и закрываются соответствующие задачи. Преподавателю сдаётся ссылка на версию (коммит или тег) и краткий отчёт.

6. Критерии оценивания

Максимальный балл за работу — **3** (календарный план силлабуса). Баллы распределяются по трём критериям силлабуса.

Таблица 3 — Критерии оценивания текущего задания

Критерий	Доля	Что проверяется
Полнота и соответствие требованиям задания	40%	все пункты задания и индивидуального варианта выполнены
Техническая корректность	40%	решения обоснованы, расчёты и код верны, терминология точна
Проверяемость материалов	20%	артефакты в репозитории, ссылки, команды запуска, данные

Работа сдаётся через канал, указанный преподавателем; в командном проекте каждый участник указывает свой вклад. Использование генеративного ИИ раскрывается по правилам университета.

7. Контрольные вопросы

1. Как обеспечивается инвариант на разных уровнях?
2. Зачем транзакция при проверке и записи?
3. Что такое идемпотентность и Idempotency-Key?
4. Почему порядок проверок — часть контракта?
5. Какие изменения контракта обратно совместимы?

Литература

1. Bass L., Clements P., Kazman R. Software Architecture in Practice. — 4th ed. — Addison-Wesley, 2021.
2. IEEE Computer Society. SWEBOK Guide v4.0a.
3. OpenAPI Initiative. OpenAPI Specification v3.1. — spec.openapis.org.