

BACKEND DEVELOPER

Цель теста: проверить навыки проектирования API, работу с БД и бизнес-логику, характерную для УСУ.

Задача:

Реализовать **REST API** для учёта контейнеров и зон хранения.

Функционал:

1. Модели:

- containers: id, number, type, status, zone_id, arrival_time
- zones: id, name, capacity, current_load, type

2. Эндпоинты:

- GET /containers — список всех контейнеров
- POST /containers — добавить контейнер
- PATCH /containers/:id — обновить статус
- GET /zones — список зон
- POST /zones/:id/assign — разместить контейнер в зону (с проверкой capacity)

3. Бизнес-логика:

- При добавлении контейнера — current_load увеличивается
- При отгрузке — current_load уменьшается
- При переполнении зоны — возвращать ошибку 400 (“Zone Overloaded”)

Требования к стеку:

- Node.js (NestJS / Express.js)
- PostgreSQL (через Prisma / TypeORM / Sequelize)
- Структура: controller → service → repository

Бонусы:

- WebSocket-события при обновлении контейнера
- Swagger-документация /docs
- Docker-compose для запуска с БД

Ожидаемый результат:

- Ссылка на GitHub
- Инструкция по установке и запуску (README.md)
- (по желанию) пример .env или Postman Collection

Срок выполнения: 1–2 дня