

## Тестовое задание

### Junior Plus Backend-разработчик (Node.js / TypeScript)

#### 1. Контекст

Мы разрабатываем платформу для совместной работы на виртуальных досках. Каждая доска состоит из «стикеров» (sticky notes) с коротким текстом. Пользователи могут создавать, редактировать и удалять стикеры в реальном времени, видя изменения друг друга мгновенно.

#### 2. Цель задания

Создать минимальный, но хорошо-структурированный прототип модуля «Стикеры», который:

1. **Обеспечивает CRUD-операции** через REST-API.
2. **Рассылает события** об операциях всем подключённым клиентам по WebSocket.
3. **Демонстрирует** ваш подход к архитектуре, стилю кода и его эффективности.

#### 3. Технические требования Технические требования

Область	Требование
Язык	TypeScript (ESM-модули)
Сервер	Express
БД	PostgreSQL (рекомендуется использование Sequelize)
Обмен данными	socket.io
Структура	Монолит. Классовый стиль
Код-стайл	ESLint + Prettier
Git	Чёткая история коммитов (Semantic Commits)

#### 4. Дополнительный необязательный функционал

Большим плюсом во время оценки работы будет наличие добавленных от Вас возможностей, которые, по-вашему, реально пригодились бы продукту в рамках текущей предметной области. Например:

- Порядок элементов с уникальным номером слоя и их перемещение (вверх / вниз относительно всех и относительно друг друга)
- Unit-тесты (Jest или Vitest)
- Аутентификация (JWT)
- И другие

Добавьте одну или несколько на ваше усмотрение.

**Важно:** возможность не должна быть поверхностной или тривиальной (например, добавление нового поля или фильтра). Она должна требовать небольшого проектирования.

## 5. Что не нужно делать

- Frontend-часть
- CI-CD / Docker

## 6. Формат сдачи

- Публичный Git-репозиторий **или** архив
- Экспорт коллекции запросов Insomnia
- Файл README.md с шагами запуска

**Важно:** Мы будем смотреть не только на финальный результат, но и на ваш ход мыслей, отражённый в коде. Задание специально имеет открытые рамки в реализации, а многие детали не уточняются. Удачи!