

VICTORY group

Техническое задание на вакансию Разработчик | Backend Developer (PHP)

Разработать RESTful API для управления задачами (To-Do List) с аутентификацией на чистом PHP.

Требования к функционалу:

1. Аутентификация:

- Регистрация пользователя (email, пароль).
- Авторизация через JWT (можно использовать библиотеку firebase/php-jwt).
- Выход из системы с инвалидацией токена.

2. Управление задачами (CRUD):

- Создание задачи: название, описание, статус ("в работе", "завершено", "дедлайн").
- Просмотр списка задач с пагинацией.
- Редактирование и удаление задач.
- Просмотр деталей задачи.

3. Валидация:

- Проверка входных данных (email, пароль, поля задач) через кастомные валидаторы.
- Форматированные JSON-ответы с HTTP-кодами (200, 400, 401, 404 и т.д.).

4. Безопасность:

- Защита от SQL-инъекций через подготовленные запросы (PDO).
- Хеширование паролей (password_hash()).
- Ограничение доступа: пользователь может работать только со своими задачами.

Технические требования:

- Язык: PHP 8.1+.
- База данных: MySQL/PostgreSQL (использовать PDO для подключения).
- Архитектура:
 - а) чистая архитектура с разделением слоев (роутер, контроллеры, сервисы, репозитории);
 - б) роутинг через .htaccess или кастомный роутер на PHP;
 - в) автозагрузка классов через Composer (PSR-4).

Документация:

- Описание API в формате OpenAPI (Swagger) или Markdown.
- Примеры запросов/ответов для всех эндпоинтов.

Деплой:

- Инструкция по настройке веб-сервера (Apache/Nginx) и БД.
- Docker-контейнер (опционально, но будет плюсом).

Дополнительные плюсы:

- Кэширование часто запрашиваемых данных (списка задач).
- Логирование ошибок и действий пользователя.
- Валидация через DTO (Data Transfer Objects).
- Фильтрация по статусу и дедлайну.
- Сортировка по дате создания/дедлайну.

Критерии оценки:

- Работоспособность: Все эндпоинты работают согласно требованиям.
- Качество кода: Отсутствие дублирования, читаемость, использование ООП.
- Безопасность: Нет уязвимостей в SQL-запросах, корректная работа с JWT.
- Документация: Полнота и понятность для пользователя API.

Что предоставить:

- Ссылку на Git-репозиторий с кодом.
- Инструкцию по установке и запуску.
- Документацию API.
- Срок выполнения: 2-3 дней.

Пояснения:

Разрешено использовать Composer для подключения библиотек (например, JWT).

Запрещено использовать Laravel, Symfony и другие фреймворки.

Акцент на самостоятельной реализации архитектуры, валидации и безопасности.

На фронтенд часть акцент не делаем.

Задание проверяет умение работать с чистым PHP, проектировать архитектуру, писать безопасный код — ключевые навыки для разработчика.

Желаем успехов в выполнении тестового задания!