# **Тестовое задание на вакансию Laravel разработчика (skill junior)**

## **Стек технологий**

* Laravel 8
* PHP >=7.3
* composer
* git
* SQL
* PHPDoc

Цель

Разработать сайт, на котором зарегистрированные пользователи смогут публиковать, редактировать и удалять статьи.

## **Требования**

1. Все посетители сайта могут просматривать опубликованные статьи.
2. Зарегистрированный пользователь может управлять только своими статьями.
3. Пользователи с правами админа могут управлять всеми статьями.
4. Каждая статья может относиться к нескольким категориям.
5. Реализовать регистрацию пользователей с подтверждением через email.
6. Кабинет пользователя:
7. Профиль.
8. Управление своими статьями.
9. Админка:
10. Управление пользователями (вкл/откл блокировки, удаление).
11. Удаление и снятие с публикации статей.
12. Управление категориями.
13. Поиск по сайту.
14. При клике на автора статьи должны выводиться все его публикации.

## **Порядок разработки**

1. Создать репозиторий на<https://gitlab.poravinternet.ru>
2. Развернуть новый проект Laravel 8
3. Сделать init commit.
 Далее коммит после выполнения каждого пункта.
4. Создать модели ( и их связи) и миграции. Не бойся что-нибудь упустить, позже новыми миграциями можно будет всё исправить (но только новыми, не меняя те, которые уже созданы)
5. Реализовать регистрацию с подтверждением через email
6. Реализовать управление пользователями и доступами (используйте spatie/laravel-permission).
7. Создать сидеры и заполнить базу случайными данными.
8. Сделать вывод списка статей (заголовок, дата, автор, категории, анонс) с постраничной разбивкой.
9. Отобразить список категорий на сайте и сделать навигацию по категориям.
10. Сделать страницу статьи.
11. Сделать поиск по сайту с постраничным выводом результатов.
12. Сделать кабинет пользователя.
13. Сделать админку.

## **Результат**

Рабочий проект, база собирается из миграций, запускаются сидеры, которые наполняют базу контентом.

При оценке результатов будем смотреть на:

1. Соответствие заданию.
2. Стиль кода.
3. Документирование.
4. ООП.
5. Описание коммитов и работа с Git.
6. Внешний вид frontend и backend приложений.
7. Удобство использования.