Техническое задание на разработку веб-приложения «Платформа для изучения иностранных языков»

1. Введение

1.1 Цель документа

Цель данного документа — определить требования к разработке веб-приложения «Платформа для изучения иностранных языков», предназначенного для упрощения процесса обучения, взаимодействия между преподавателями и студентами, а также для управления образовательным контентом.

1.2 Актуальность и назначение проекта

Современные технологии предоставляют широкие возможности для онлайн-обучения, что особенно актуально в условиях удаленной работы и учебы. Разрабатываемая платформа будет способствовать повышению доступности качественного обучения языкам, предоставляя инструменты для управления уроками, тестами и анализом прогресса студентов.

1.3 Основные пользователи системы

Администраторы — управление пользователями, курсами и аналитикой.

Преподаватели — создание курсов, уроков и тестов, а также управление отзывами.

Студенты — прохождение уроков, выполнение тестов, анализ прогресса.

2. Цели и задачи проекта

2.1 Основные цели

Обеспечение удобного процесса изучения иностранных языков через централизованную платформу.

Упрощение управления курсами и студентами для преподавателей.

Предоставление инструментов для анализа прогресса обучения.

2.2 Конкретные задачи

Разработка интерфейса для просмотра и прохождения курсов.

Реализация функционала для создания и управления образовательным контентом.

Создание системы тестирования знаний студентов.

Внедрение аналитических инструментов для администраторов и преподавателей.

Пример: Преподаватель должен иметь возможность добавлять курсы с уроками различных форматов (текст, видео, аудио), а студенты — проходить их, отслеживать прогресс и оставлять отзывы.

3. Требования к системе

3.1 Функциональные требования

Управление пользователями:

Регистрация, авторизация и управление профилями пользователей.

Разграничение прав доступа: администраторы, преподаватели, студенты.

Курсы и уроки:

Добавление, редактирование и удаление курсов.

Управление уроками с различными типами контента (текст, видео, аудио).

Тестирование:

Создание тестов с вариантами ответов.

Автоматическая проверка тестов.

Отзывы:

Возможность студентам оставлять отзывы с рейтингами для курсов.

Модерация отзывов преподавателями.

Статистика и отчеты:

Преподаватели могут отслеживать прогресс студентов.

Администраторы видят общую аналитику по системе.

Пример: При прохождении урока студент должен видеть индикатор прогресса и возможность повторного прохождения теста.

3.2 Нефункциональные требования

Производительность: время отклика интерфейса не более 2 секунд.

Масштабируемость: поддержка до 10,000 пользователей одновременно.

Безопасность:

Шифрование данных через SSL.

Хранение паролей в хешированном виде.

4. Требования к пользовательскому интерфейсу

4.1 Основные экраны

Главная страница: отображение доступных курсов с фильтрацией по уровню сложности.

Страница курса: список уроков, прогресс, отзывы и тесты.

Административная панель: управление пользователями, курсами и аналитика.

Личный кабинет студента: пройденные уроки, тесты и отзывы.

4.2 Навигация

Удобное меню с основными разделами: «Курсы», «Тесты», «Прогресс», «Настройки».

4.3 Юзабилити

Адаптивный дизайн для настольных устройств.

Интуитивно понятный интерфейс с минимальным количеством действий для выполнения задач.

Пример: На странице курса студент должен видеть прогресс по урокам и возможность начать следующий урок.

5. Требования к технической реализации

5.1 Языки и технологии

Backend: Laravel.

Frontend: Чистый JavaScript или Vue.js/React, AJAX-запросы через jQuery.

База данных: MySQL.

5.2 Архитектура системы

REST API для взаимодействия между клиентом и сервером.

Модульная архитектура с разделением логики.

5.3 Интеграции

Внешние уведомления через email (например, SendGrid).

6. Требования к безопасности

6.1 Аутентификация и авторизация

Разделение прав доступа.

6.2 Шифрование данных

HTTPS для всех соединений.

Хранение паролей через bcrypt.

6.3 Политики доступа

SQL-инъекции и XSS должны быть предотвращены.

Ограничение по IP для административного доступа.

7. Ограничения и допущения

7.1 Технические ограничения

7.2 Сроки выполнения

Разработка — месяц.

Тестирование — хз.

8. Требования к тестированию и приемке

8.1 Типы тестирования

Функциональное: проверка всех функций.

Тестирование безопасности: защита от уязвимостей.

8.2 Критерии приемки

Все требования выполнены.

9. Требования к документации

9.1 Пользовательская документация

Инструкции по регистрации, прохождению курсов и тестов.

9.2 Техническая документация

Архитектура, структура базы данных, API.