

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)

Кафедра физики и прикладной математики

ЛАБОРАТОРНАЯ РАБОТА № 1

по дисциплине: «Технологии разработки веб-приложений»

на тему: «Проектирование архитектуры и макетов веб-приложения»

Выполнил:
студент группы ИТ-122
Минаев Г. В.

Проверил:
ассистент кафедры ФиПМ
Болачков А. В.

Цель работы: развитие навыков разработки и согласования технических спецификаций на веб-приложения.

Задание: создать рабочее пространство для выполнения лабораторных работ и курсовой работы. Спроектировать и задокументировать информационную и программную архитектуру веб-приложения, разработать макеты страниц.

Выполнение работы:

Была выбрана тема «Использование искусственного интеллекта в языковом обучении», потому что использование искусственного интеллекта в обучении иностранным языкам открывает новые возможности для улучшения качества образования. Персонализация учебного процесса с помощью ИИ позволяет каждому студенту учиться в собственном темпе и получать задания, которые соответствуют его уровню знаний. Это особенно актуально в современном мире, где глобализация требует высокого уровня владения иностранными языками. Кроме того, автоматизация рутинных задач, таких как проверка заданий и создание учебных материалов, значительно упрощает работу преподавателей, позволяя им сосредоточиться на методической и практической помощи студентам.

Веб-приложение предназначено для повышения эффективности изучения иностранных языков за счет использования искусственного интеллекта (ИИ). Приложение помогает студентам осваивать язык, а преподавателям — автоматизировать рутинные задачи. Цель — предложить персонализированный подход к обучению, учитывая уровень знаний, стиль обучения и потребности пользователя.

Описание предметной области:

- Тестирование на уровень знаний

Пользователи проходят вступительный тест, который определяет их уровень владения языком по международной шкале (A1–C2). На основе результатов система предлагает дальнейший учебный маршрут.

- Создание индивидуального курса

После тестирования ИИ разрабатывает индивидуальную программу обучения, учитывающую сильные и слабые стороны пользователя. Программа может включать задания на расширение словарного запаса, грамматические упражнения, аудирование и разговорные тренажёры. Этот курс адаптируется по мере прогресса студента.

- Разработка различных типов заданий

Приложение предоставляет множество типов заданий: тесты с выбором ответа, упражнения на грамматику, перевод текста и диалоги с виртуальным ассистентом. ИИ помогает настроить сложность заданий в зависимости от уровня знаний пользователя.

- Поддержка преподавателей

Преподаватели могут использовать ИИ для создания уникальных заданий, тестов и упражнений. Приложение автоматически генерирует предложения по заданиям на основе учебных целей. Система анализирует прогресс учеников и помогает преподавателям корректировать учебные планы.

- Автоматическая оценка

Приложение автоматически оценивает тесты и письменные задания, анализирует ошибки и предоставляет пользователю обратную связь. Преподаватели могут сэкономить время на проверке заданий и сосредоточиться на обучении студентов.

- Учебные материалы

В зависимости от результатов тестирования и текущего уровня, пользователи получают доступ к теоретическим материалам, соответствующим их знаниям. Это могут быть грамматические правила, примеры использования слов и выражений, а также тематические тексты для закрепления материала. Теоретические материалы структурированы по уровням (A1-C2) и направлены на улучшение базовых и продвинутых навыков студентов.

Ссылка на репозиторий:

<https://github.com/georgy006/LearnLand>

Описание функциональных требований к приложению

1. Регистрация и аутентификация пользователей

- Пользователь должен иметь возможность зарегистрироваться в системе, указав личные данные и уровень знания языка.
- Пользователь должен иметь возможность входить в систему с использованием своих учетных данных.

2. Тестирование на уровень знаний

- Пользователь должен пройти тест для определения уровня знаний.
- Система должна предоставлять результаты теста с указанием уровня (A1-C2).

3. Создание индивидуального курса

- Система должна генерировать персонализированный учебный план на основе результатов тестирования.
- Пользователь должен иметь возможность просматривать и изменять свой курс.

4. Разработка и выполнение заданий

- Пользователь должен иметь доступ к различным типам заданий (тесты, упражнения, аудирование).
- Система должна оценивать выполненные задания и предоставлять обратную связь.

5. Поддержка преподавателей

- Преподаватель должен иметь возможность создавать и редактировать задания.
- Система должна предоставлять аналитику о прогрессе студентов.

6. Доступ к теоретическим материалам

- Пользователь должен иметь доступ к теоретическим материалам, соответствующим его уровню.

7. Чат с ИИ для оттачивания знаний

- Пользователь должен иметь возможность взаимодействовать с ИИ в режиме чата для оттачивания языковых навыков.

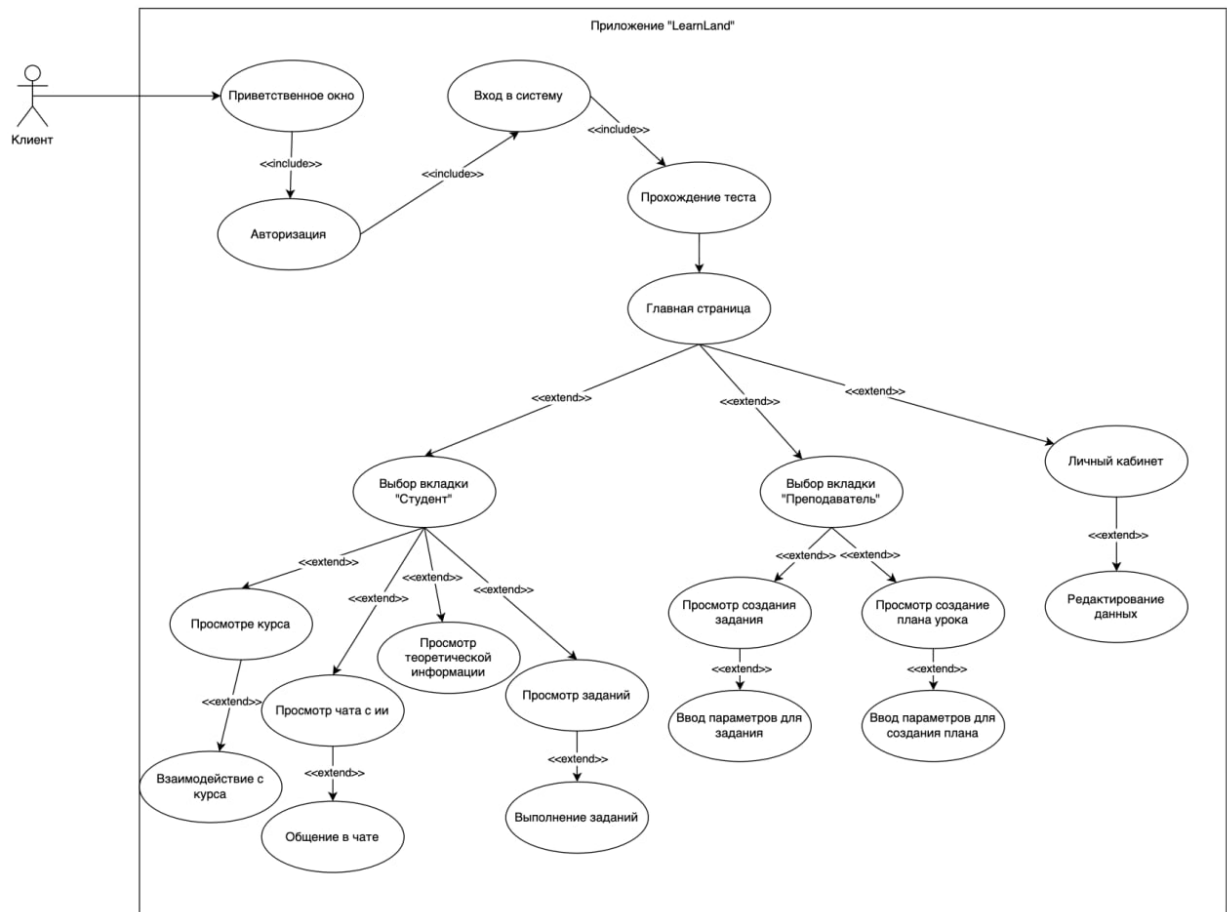


Рисунок 1 – Uml-диаграмма веб-приложения

Информационная структура приложения.

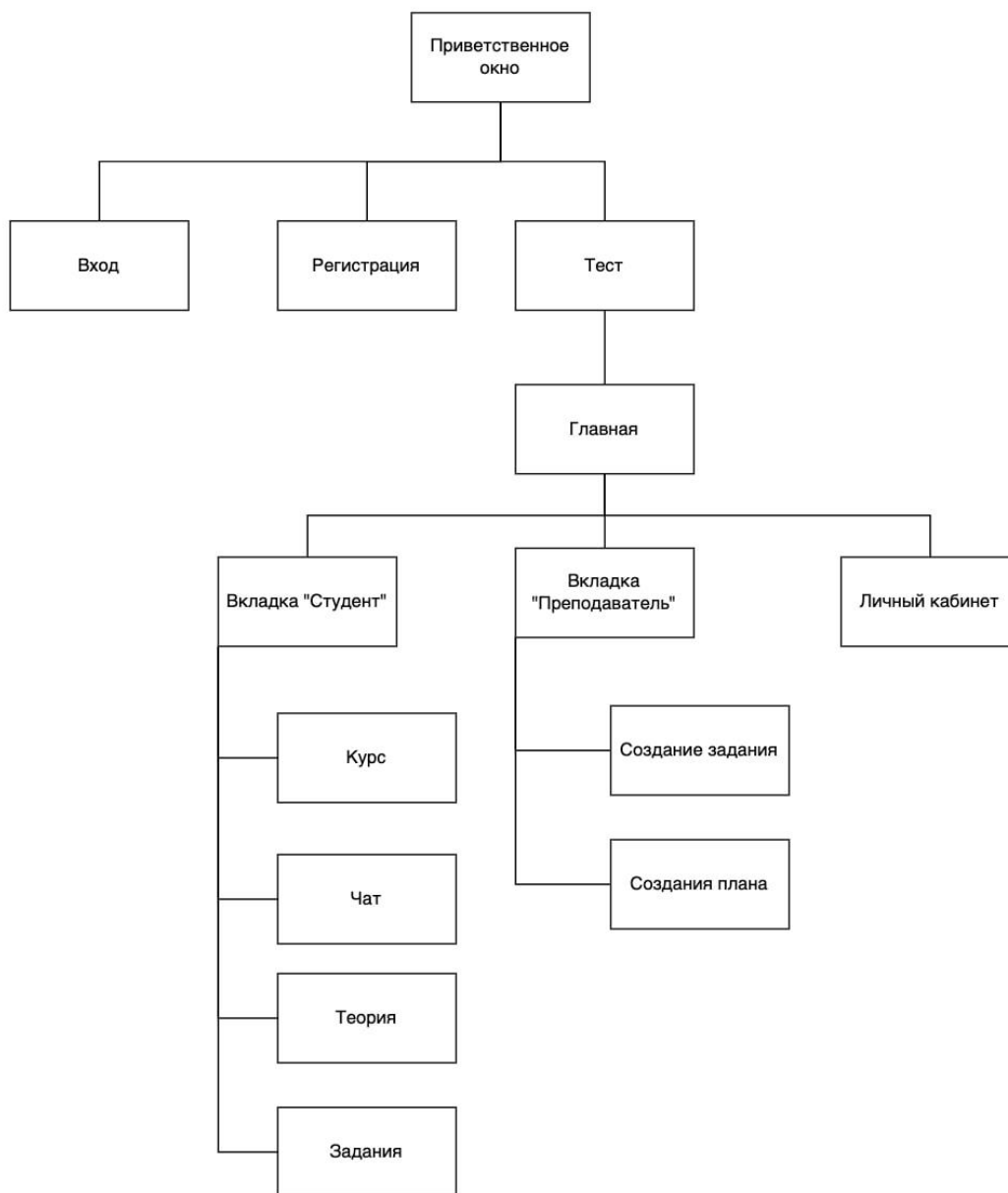


Рисунок 2 – Информационная архитектура веб-приложения.

Структура интерфейса

1. Приветственное окно

При запуске приложения пользователи попадают на приветственное окно, откуда они могут перейти к входу или регистрации.

2. Вход

Пользователи могут войти в свою учетную запись, чтобы получить доступ к персонализированному контенту.

3. Регистрация

Новый пользователь может зарегистрироваться, заполнив необходимые данные для создания учетной записи.

4. Тест

Приложение предлагает пройти тест, который поможет определить уровень знаний пользователя и адаптировать последующее обучение.

5. Главная страница

После входа в систему пользователи попадают на главную страницу, где могут выбрать соответствующую вкладку в зависимости от своей роли (студент или преподаватель).

6. Вкладка "Студент"

Студенты могут просматривать курсы, которые они изучают, и отслеживать свой прогресс.

Встроенный чат позволяет студентам взаимодействовать с ии как с носителем языка

Раздел с теоретическими материалами, которые студенты могут изучать в рамках курсов.

Студенты могут просматривать и выполнять задания, поставленные преподавателем, для проверки своих знаний.

7. Вкладка "Преподаватель"

Преподаватели могут создавать и настраивать задания для студентов, определяя их содержание и параметры.

Преподаватели могут разрабатывать планы уроков, включая необходимые теоретические материалы и задания.

8. Личный кабинет

В этом разделе пользователи могут редактировать свои данные и управлять настройками учетной записи.

Программная архитектура веб-приложения

1. Клиентская часть (Frontend)

Технологии: HTML, CSS, JavaScript (TypeScript), Angular.

Описание: Клиентская часть веб-приложения будет реализована с использованием Angular, что позволит создать интерактивный интерфейс для пользователей.

Вся функциональность будет удобно представлена пользователю через интуитивно понятный интерфейс с возможностью быстрой навигации между разделами.

2. Серверная часть (Backend)

Технологии: Java, Spring Boot, PostgreSQL.

Описание: Серверная часть будет основана на Java с использованием фреймворка Spring Boot, который обеспечит быстрый и стабильный бэкенд для обработки запросов от клиента. Важные задачи серверной части включают:

Обработка запросов от клиентской части через REST API.

Взаимодействие с базой данных для хранения курсов, заданий, пользовательских данных, а также результатов выполнения тестов.

Обеспечение логики персонализированных курсов на основе тестирования и прогресса студентов.

Поддержка чат-ботов (ИИ и ИИ-носителя) для взаимодействия с пользователями.

3. База данных

Технологии: PostgreSQL.

Описание: Реляционная база данных будет использоваться для хранения данных о пользователях, курсах, тестах, заданиях, планах уроков, а также переписок в чатах. Все данные будут структурированы таким образом, чтобы поддерживать масштабируемость и эффективность обработки запросов.

4. Взаимодействие с внешними сервисами

ChatGPT API: будет использоваться для интеграции ИИ-помощника, который будет отвечать на вопросы студентов и преподавателей, а также предоставлять задания и помогать в изучении материала.

REST API: будет создан для связи клиентской части с серверной частью и взаимодействия с базой данных, а также для интеграции с ИИ-носителями.

5. Архитектурные особенности

Микросервисная архитектура: Каждый компонент приложения (аутентификация, курсы, чаты, тесты и задания) будет разделен на отдельные сервисы, что позволит повысить гибкость и масштабируемость системы.

Безопасность: будут реализованы меры безопасности, включая аутентификацию и авторизацию пользователей, с использованием JWT (JSON Web Token).

Макеты страниц

Были спроектированы прототипы отдельных страниц веб-приложения:

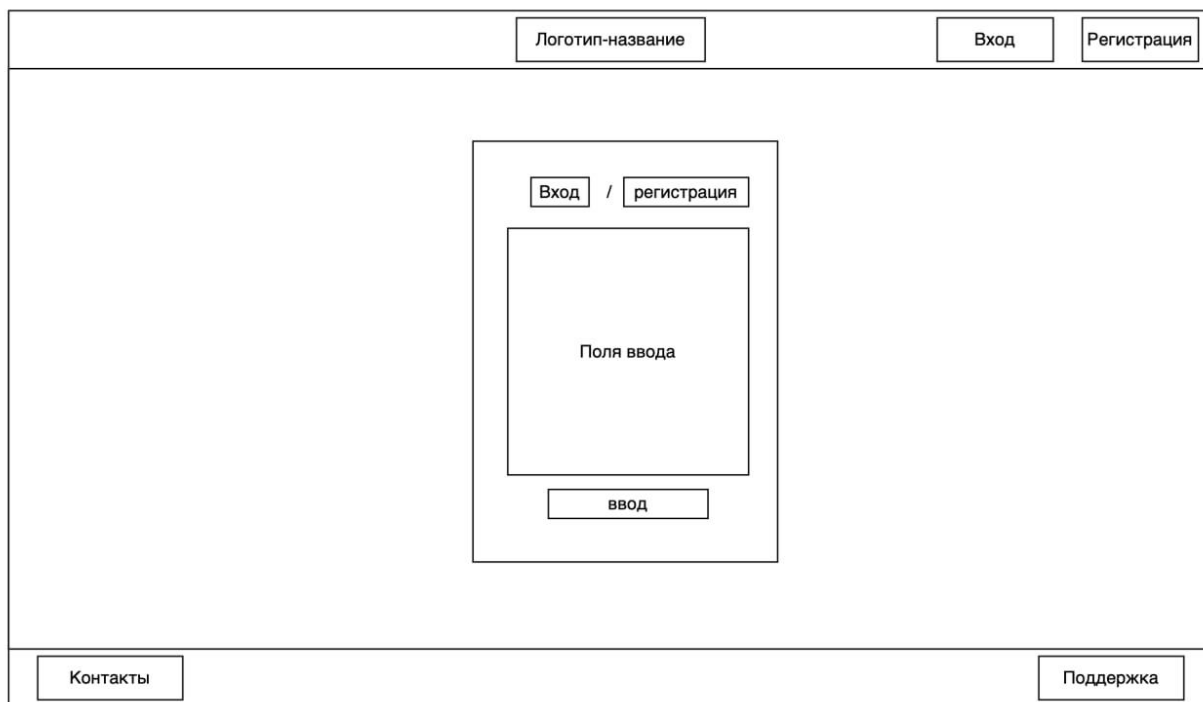


Рисунок 3 – Регистрация и вход

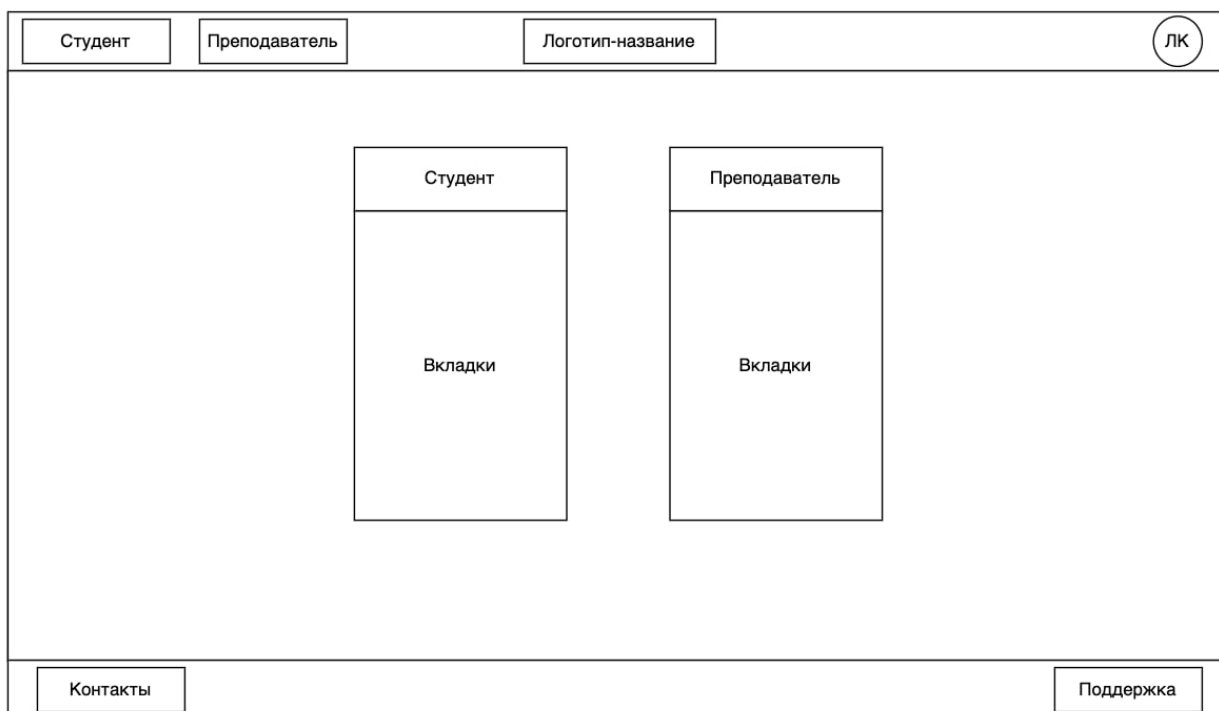


Рисунок 4 – Главная страница



Рисунок 5 – Приветственная страница



Рисунок 6 – Личный кабинет

Логотип-название		ЛК
Вопрос		
<div>Ответ1</div> <div>Ответ1</div> <div>Ответ1</div> <div>Ответить</div>		
Контакты	Поддержка	

Рисунок 7 – Тестовое задание

Студент	Преподаватель	Логотип-название	ЛК
<div>Студент</div> <div>Курс</div> <div>Чат</div> <div>Теория</div> <div>Задания</div>	<div>Основное содержание</div>		
Контакты	Поддержка		

Рисунок 8 – окно «Студент»

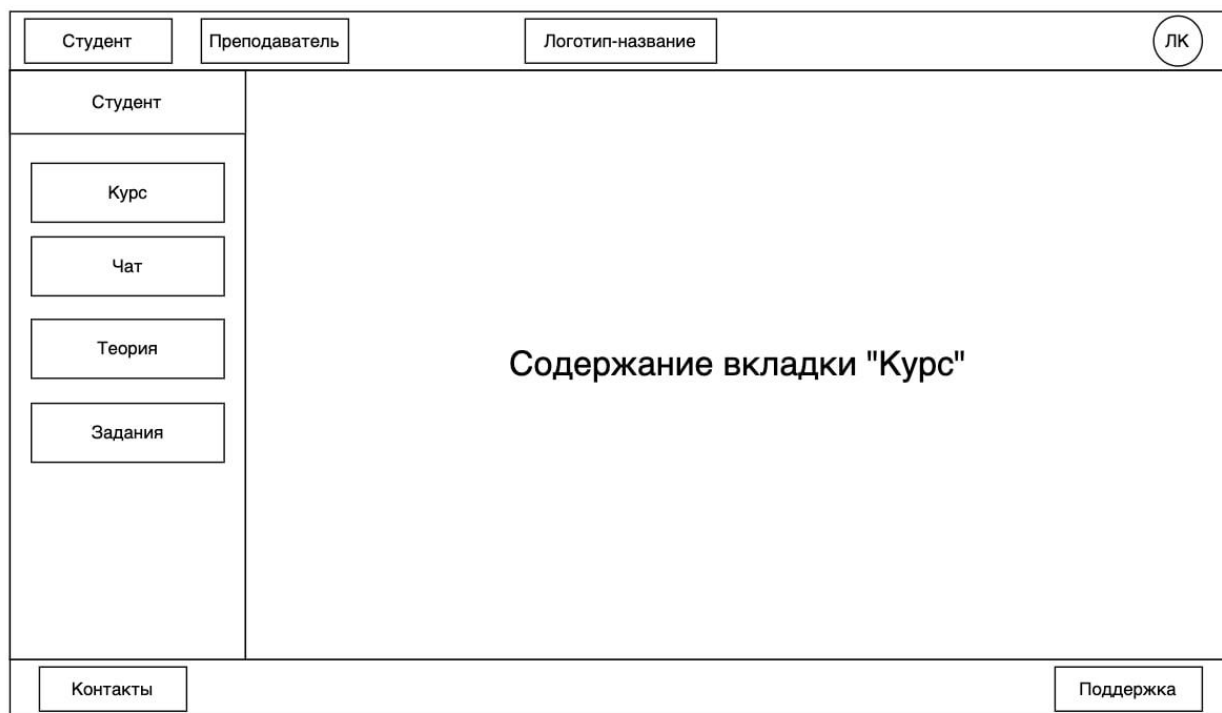


Рисунок 9 – вкладка курса

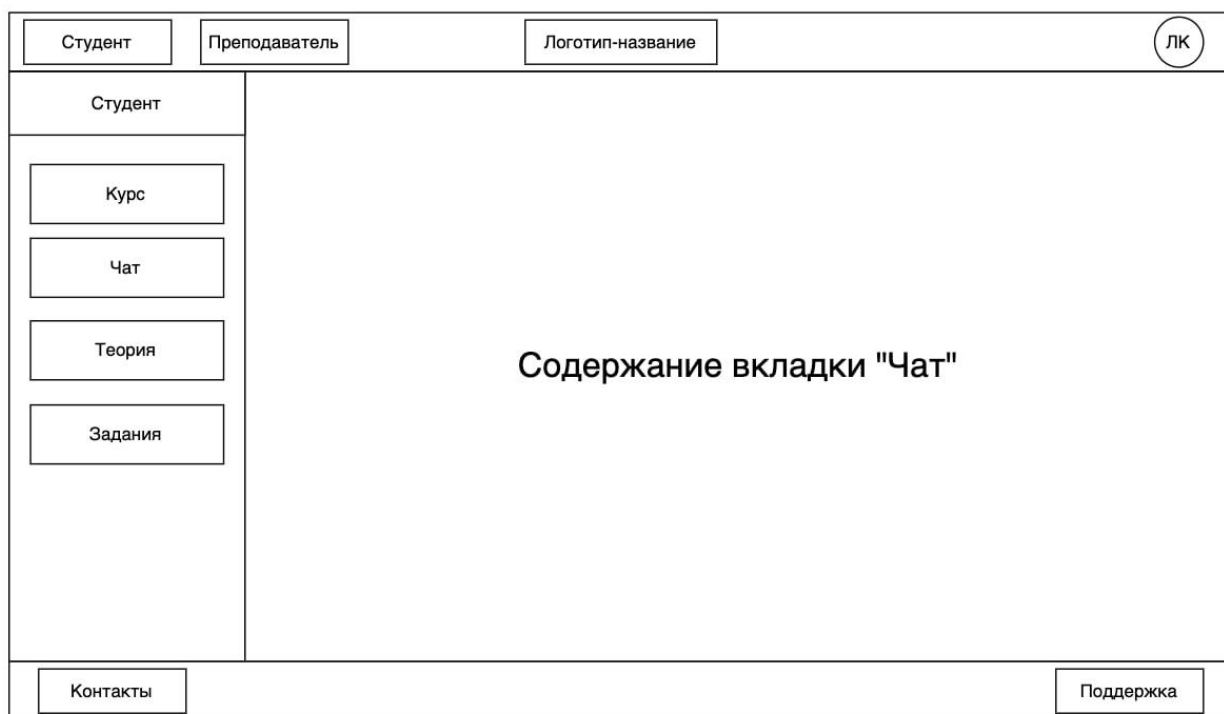


Рисунок 10 – вкладка чата



Рисунок 11 – вкладка с теорией

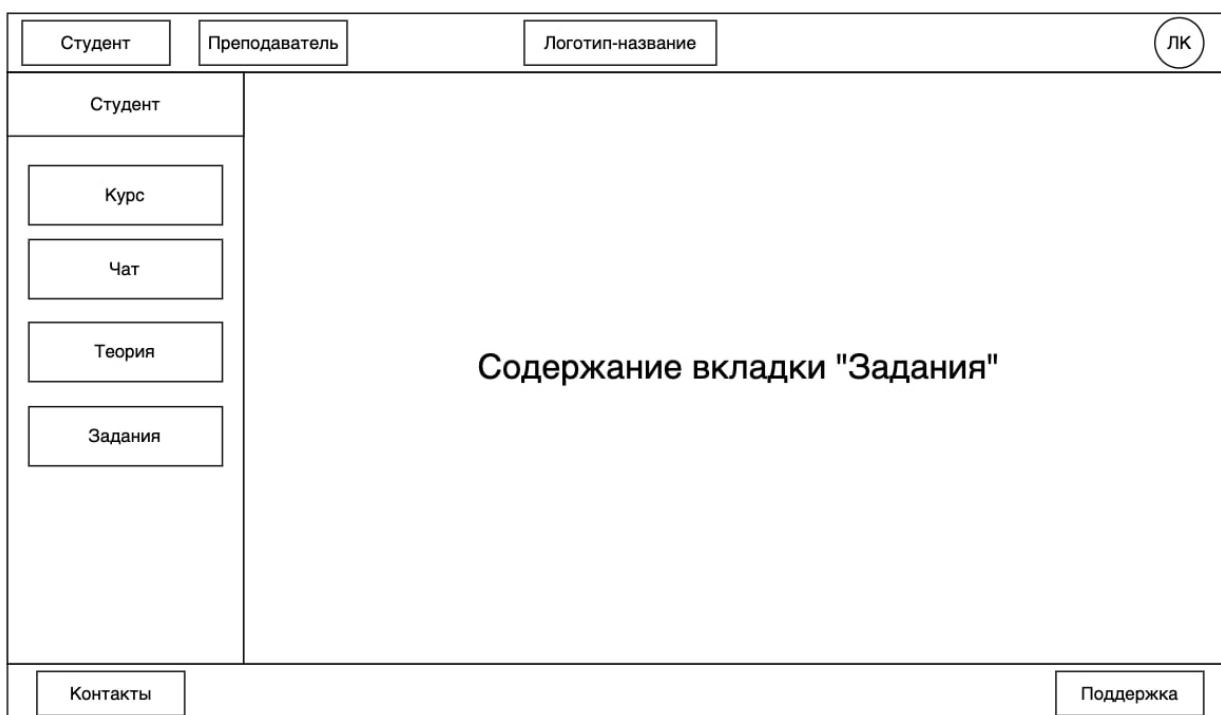


Рисунок 12 – вкладка с заданиями



Рисунок 13 – окно «Преподаватель»

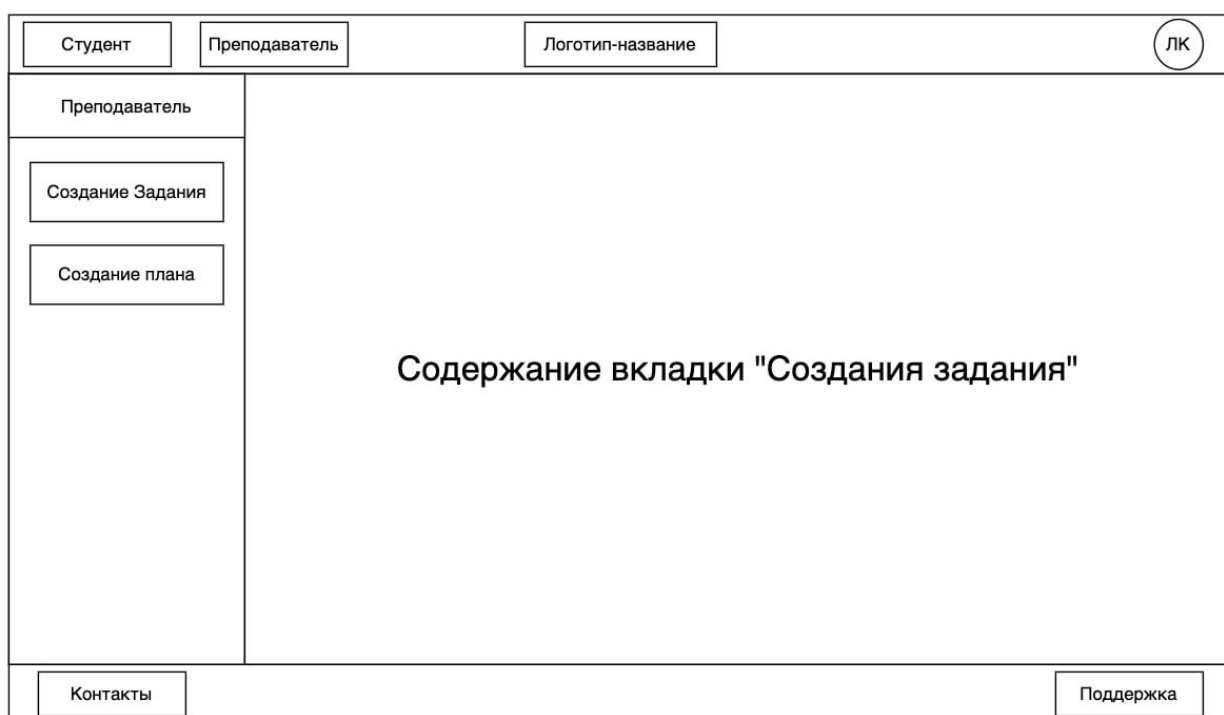


Рисунок 14 – вкладка с генератором заданий

Студент	Преподаватель	Логотип-название	ЛК
Преподаватель	<div>Содержание вкладки "Создание плана урока"</div>		
Создание Задания			
Создание плана			
Контакты		Поддержка	

Рисунок 15 – вкладка с генератором планом урока

Вывод: Результаты данной работы показывают успешную интеграцию ИИ в образовательную среду для изучения иностранных языков. Приложение "LearnLand" не только улучшает учебный процесс студентов, но и облегчает работу преподавателей, предлагая инструменты для автоматизации задач и персонализированного обучения. Тщательное проектирование архитектуры и документация дают четкую основу для дальнейшей разработки и внедрения этого решения в реальных образовательных системах.

Работа продемонстрировала успешное овладение навыками проектирования веб-приложений и документирования программной архитектуры, а также подтвердило актуальность применения современных технологий в образовании.