

Хранение данных проекта требует надежной и масштабируемой системы управления базами данных. В качестве СУБД выбрана PostgreSQL, поскольку она поддерживает сложные запросы, масштабируемость и обеспечивает высокую надежность хранения данных. Дополнительно планируется использование Redis для кэширования и ускорения обработки запросов.

Таким образом, разработка веб-приложения для управления проектами с применением ИИ позволит автоматизировать рутинные процессы, повысить эффективность работы команд и сократить временные затраты на принятие решений. Основная миссия проекта – создание интеллектуального помощника, который будет анализировать состояние проекта и предлагать оптимальные пути его выполнения. Внедрение ИИ способствует повышению точности прогнозов, улучшению распределения ресурсов и, как следствие, увеличению успешности реализуемых проектов.

Е. В. Киптик, Д. С. Кузьменков

(ГГУ имени Ф. Скорины, Гомель)

РАЗРАБОТКА BACK-END ЧАСТИ ПРИЛОЖЕНИЯ «ИНТЕРАКТИВНАЯ ДОСКА» С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА ПРОГРАММИРОВАНИЯ C#

«Интерактивная доска» – это современное приложение для совместной работы в реальном времени, позволяющее пользователям создавать, редактировать и взаимодействовать с цифровым контентом: текстовыми заметками, графиками, изображениями и мультимедиа. Оно поддерживает многопользовательский режим, где несколько участников могут одновременно работать над проектом, видя изменения друг друга в реальном времени. Приложение идеально подходит для удаленных команд, образовательных учреждений и творческих коллективов.

Для разработки выбраны технологии, обеспечивающие высокую производительность, гибкость и масштабируемость. ASP.NET используется для обработки HTTP-запросов и управления серверной частью, SignalR обеспечивает двустороннее общение в реальном времени для синхронизации действий пользователей, таких как ри-

сование или добавление текста. Entity Framework Core упрощает работу с базой данных через ORM, а MongoDB хранит неструктурированные данные, такие как JSON-объекты. Redis ускоряет доступ к часто используемым данным через кэширование, Swagger (OpenAPI) применяется для документирования и тестирования API.

Микросервисная архитектура back-end позволяет разбить систему на независимые сервисы, каждый из которых отвечает за конкретную бизнес-логику. Это обеспечивает масштабируемость, гибкость и устойчивость к сбоям. Например, сервис аутентификации управляет пользователями и их правами, SignalR синхронизирует действия в реальном времени, MongoDB хранит данные о компонентах доски, а Redis кэширует часто используемую информацию. Такая архитектура упрощает тестирование, развертывание и оптимизацию ресурсов, позволяя эффективно распределять нагрузку.

Использование микросервисной архитектуры делает back-end приложения более адаптивными к изменениям и нагрузкам, что особенно важно для такого динамичного продукта, как интерактивная доска.

А. В. Кирчук, В. С. Закревская
(ГГУ имени Ф. Скорины, Гомель)

ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ GODOT ENGINE ПРИ СОЗДАНИИ 2D-ИГР

При поиске оптимальных технологий для реализации проекта по созданию 2D игры был проведен небольшой анализ современных игровых движков с целью выявления как преимуществ, так и недостатков каждого.

Для проверки по заданным параметрам выбраны наиболее популярные на сегодняшний день игровые движки, а именно: Unreal Engine, Unity, Godot Engine, Cocos2d и GameMaker Studio.

В результате проведенного исследования был выбран Godot Engine, так как он обладает многими преимуществами, основные из которых приведем ниже: