

ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ ФРИЛАНС ПЛАТФОРМЫ НА ОСНОВЕ МИКРОСЕРВИСНОЙ АРХИТЕКТУРЫ

Рабец И.О., студент

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Жвакина А.В. – канд. техн. наук, доцент

Аннотация. В работе представлено веб-приложение для автоматизации взаимодействия заказчиков и исполнителей в сфере фриланс-услуг. Для разработки использованы: ASP.NET, Angular, Docker, REST/gRPC, брокер сообщений, что обеспечивает масштабируемость, устойчивость к нагрузкам и безопасность данных. Особенностью системы является интеграция механизмов рейтингования, отзывов и прозрачного контроля выполнения задач, направленных на снижение временных затрат и повышение доверия участников цифрового рынка.

В условиях активного развития цифровой экономики и роста популярности удалённой занятости всё большую значимость приобретают информационные системы, позволяющие эффективно организовать взаимодействие между заказчиками и исполнителями. Одним из таких решений стало разработанное веб-приложение, ориентированное на автоматизацию процессов в сфере фриланс-услуг и проектной деятельности.

Разработанная система представляет собой современное веб-приложение, предназначенное для упрощения и оптимизации взаимодействия между заказчиками и исполнителями. Программный продукт реализует комплексный подход к организации цифровой среды, в которой пользователи могут быстро и легко находить друг друга, формировать задачи, координировать процесс выполнения работ и осуществлять взаиморасчёты в рамках единой платформы. В основе системы лежит идея объединения ключевых бизнес-процессов фриланс-рынка в одном информационном пространстве с акцентом на удобство, гибкость и возможность дальнейшего развития.

Функциональные возможности системы охватывают полный жизненный цикл взаимодействия участников и включают:

- регистрацию и аутентификацию пользователей с последующим управлением профилями;
- публикацию проектов и подачу заявок на выполнение работ;
- поиск и фильтрацию задач и исполнителей по заданным критериям;
- обмен сообщениями и файлами между участниками в рамках проектной активности;
- управление проектами, включая отслеживание статусов задач, контроль сроков и фиксацию результатов.

Система решает ряд актуальных задач, характерных для современной цифровой экономики. Прежде всего, она позволяет снизить затраты времени и усилий при поиске исполнителей и организации проектной деятельности. Автоматизация подачи заявок, выбора кандидатов и ведения коммуникации делает работу пользователей более быстрой и удобной. Кроме того, повышается прозрачность взаимодействия за счёт внедрения механизмов рейтингов, отзывов и контроля выполнения задач. Особое внимание уделено безопасности данных и финансовых операций, что напрямую влияет на уровень доверия пользователей.

Технологическая основа системы построена с использованием современных инструментов и архитектурных подходов. Серверная часть реализована на платформе ASP.NET, что обеспечивает высокую производительность и кроссплатформенность [1]. Клиентская часть разработана с использованием фреймворка Angular, благодаря чему удалось создать удобный и понятный пользовательский интерфейс. Важной особенностью является использование микросервисной архитектуры, при которой система разделена на независимые компоненты, взаимодействующие через API. Такой подход делает систему более гибкой, устойчивой и удобной для масштабирования [2].

Для организации взаимодействия между сервисами используются как синхронные протоколы (REST, gRPC), так и асинхронные механизмы обмена сообщениями через брокер сообщений, что позволяет эффективно справляться с высокой нагрузкой. Контейнеризация с использованием Docker и Docker Compose упростила процесс развертывания и сопровождения системы. Дополнительно применяются инструменты логирования, мониторинга и тестирования, что положительно сказывается на надёжности программного продукта.

В результате разработанная система представляет собой технологически современное решение для автоматизации процессов фриланс-платформы. Она сочетает широкий функционал, удобство для пользователя и использование актуальных технологий, что делает её востребованной и конкурентоспособной на рынке цифровых услуг.

Список использованных источников:

1. ASP.NET Core. – [Электронный ресурс]. – Режим доступа : <https://dotnet.microsoft.com/en-us/apps/aspnet>.
2. Микросервисная архитектура. – Режим доступа : <https://www.atlassian.com/ru/microservices/microservices-architecture>.