**Техническое Задание на Разработку Веб-приложения «NestChat»**

**1. Введение**

1.1 **Цель документа**  
Этот документ описывает требования к разработке веб-приложения «NestChat» для обеспечения эффективной коммуникации между пользователями в реальном времени. Приложение должно поддерживать обмен сообщениями, создание чатов, отправку уведомлений и обмен файлами с использованием современных технологий.

1.2 **Актуальность и назначение проекта**  
В современном мире эффективность коммуникации и быстрота обмена информацией являются важными аспектами для любой команды и бизнеса. Система «NestChat» предоставляет удобный инструмент для общения в реальном времени, с возможностью создания групповых чатов и интеграцией с другими сервисами.

1.3 **Основные пользователи системы**  
Система предназначена для использования:

* **Пользователи** — отправка сообщений, участие в чатах, отправка файлов.
* **Администраторы** — управление пользователями и чатами, мониторинг активности.

**2. Цели и задачи проекта**

2.1 **Основные цели**

* Обеспечить пользователям возможность обмена сообщениями в реальном времени.
* Реализовать систему уведомлений для новых сообщений и активности в чатах.
* Создание групповых чатов с возможностью обмена файлами и медиа.

2.2 **Конкретные задачи**

* Разработка функционала для создания, редактирования и удаления чатов.
* Реализация системы уведомлений для новых сообщений и активности.
* Поддержка отправки текстовых сообщений, изображений и документов.
* Обеспечение работы в реальном времени с использованием WebSocket или других технологий.
* Разработка системы безопасности и защиты данных.

**3. Требования к системе**

3.1 **Функциональные требования**

* **Чаты:** возможность создания, редактирования и удаления чатов.
* **Отправка сообщений:** поддержка отправки текста, изображений, файлов.
* **Уведомления:** уведомления о новых сообщениях в чатах.
* **Реальное время:** поддержка обмена сообщениями в реальном времени с использованием WebSocket.
* **История сообщений:** возможность просматривать историю сообщений в каждом чате.

3.2 **Нефункциональные требования**

* **Производительность:** отклик интерфейса не более 2 секунд при отправке и получении сообщений.
* **Масштабируемость:** возможность поддерживать работу с 20+ пользователями без потери производительности.
* **Безопасность:** поддержка HTTPS/HTTP и защита данных через SSL/TLS, шифрование сообщений.

**4. Требования к пользовательскому интерфейсу**

4.1 **Основные экраны**

* **Главная страница:** отображение списка доступных чатов, создание нового чата.
* **Страница чата:** отображение сообщений, возможность отправки новых сообщений, прикрепления файлов, медиа и эмоций.
* **Профиль пользователя:** отображение личной информации пользователя, возможность редактирования.

4.2 **Навигация** Меню должно включать разделы:

* «Профиль»
* «Чаты»
* «Выход»

4.3 **Юзабилити**

* Простой и интуитивно понятный интерфейс для быстрого доступа к чатам и сообщениям.
* Возможность использования смайлов и эмодзи (через интеграцию с библиотекой **emoji-picker-react**).

**5. Требования к технической реализации**

5.1 **Языки и технологии**

* **Frontend:** React.js, TailwindCSS, Vite, DaisyUI, Emoji-picker-react.
* **Backend:** Laravel (в сочетании с Pusher и Laravel Echo для реального времени).
* **База данных:** MySQL или PostgreSQL.
* **Интеграции:** Платформа для WebSocket (например, **Pusher** или **Laravel Echo**), интеграция с внешними сервисами для отправки уведомлений (например, **email-сервис**).

5.2 **Архитектура системы**

* Микросервисная архитектура с использованием **Inertia** и **WebSocket** для реального времени.
* Хранение данных в реляционной базе данных (MySQL, MongoDB).

5.3 **Интеграции**

* Интеграция с сервисами уведомлений через **Pusher**.
* Интеграция с внешними сервисами для отправки email-уведомлений.

**6. Требования к безопасности**

6.1 **Аутентификация и авторизация**

* **Авторизация:** разграничение прав доступа — администраторы, пользователи, создание и администрирование чатов.

6.2 **Шифрование данных**

* Использование **HTTPS** для защиты передаваемых данных.
* Хранение паролей с использованием **bcrypt**.

6.3 **Политики доступа**

* Защита от **SQL-инъекций**, **XSS** и **CSRF-атак**.
* Ограничение доступа к административной панели по IP-адресам.

**7. Ограничения и допущения**

7.1 **Технические ограничения**

* Приложение должно быть оптимизировано для работы на облачных сервисах (например, **AWS** или **Azure**).

7.2 **Финансовые ограничения**

* Бюджет: 3 сирийские шаурмы

7.3 **Сроки выполнения**

* Полная реализация проекта — 4 месяца с начала разработки.

**8. Требования к тестированию и приемке**

8.1 **Типы тестирования**

* **Функциональное тестирование:** проверка всех функций (создание чатов, отправка сообщений, уведомления).
* **Нагрузочное тестирование:** проверка на работу с более чем 10 пользователями.
* **Тестирование безопасности:** проверка на уязвимости, включая SQL-инъекции и XSS.

8.2 **Критерии приемки**

* Полное выполнение функциональных и нефункциональных требований.
* Прохождение более 85% тестов по результатам приемочных испытаний.

**9. Требования к документации**

9.1 **Пользовательская документация**

* Руководство пользователя с инструкциями по созданию чатов, отправке сообщений и настройке профиля.

9.2 **Техническая документация**

* Описание архитектуры приложения, структура базы данных, API для взаимодействия между клиентом и сервером.

**10. План реализации**

10.1 **Этапы разработки**

1. **Анализ и проектирование** — 1 месяц.
2. **Разработка прототипа** — 1 месяц.
3. **Полноценная разработка и тестирование** — 3 месяца.
4. **Внедрение и обучение** — 1 месяц.

10.2 **Сроки выполнения этапов**

* Дата начала: 21 октября 2024 года.
* Дата завершения: 10 декабря 2024 года.

10.3 **Ответственные лица**

* Учредитель по совместительству инвестор фирмы: Ларионова А.И  
  Глава отдела HR: Малых В.А  
  Корпоративный директор заведующий отдела Back’end: Вакуров Ф.А  
  Корпоративный директор заведующий отдела Front’end: Галкин Н.А
* Ведущий разработчик: Подопригоров И.В.

**11. Приложения**

* **Приложение A:** Примеры интерфейсов.