**Техническое задание на разработку веб-приложения** «Платформа для бронирования и управления услугами салона красоты»

**1. Введение**

1.1 **Цель документа**

Этот документ описывает требования к разработке веб-приложения «Платформа для бронирования и управления услугами салона красоты», предназначенного для автоматизации процессов бронирования, управления расписанием и предоставления удобного интерфейса для клиентов, мастеров и администраторов.

1.2 **Актуальность и назначение проекта**

В условиях растущей популярности цифровых сервисов и онлайн-записи на услуги, платформа позволит улучшить клиентский опыт, оптимизировать рабочие процессы салонов красоты и упростить управление расписанием мастеров.

1.3 **Основные пользователи системы**

Система предназначена для использования сотрудниками компании:

* Администраторы — управление расписанием, учетными записями пользователей и статистикой.
* Мастера — управление собственным расписанием, просмотр записей клиентов и получение уведомлений.
* Клиенты — поиск услуг, бронирование, получение напоминаний и оставление отзывов.

**2. Цели и задачи проекта**

2.1 **Основные цели**

* Автоматизация процесса бронирования и управления услугами.
* Повышение прозрачности и удобства взаимодействия между салоном красоты и клиентами.

2.2 **Конкретные задачи**

* Разработка функционала бронирования услуг с выбором мастера, даты и времени.
* Создание модуля управления расписанием для администраторов и мастеров.
* Внедрение системы уведомлений для клиентов и сотрудников.
* Разработка аналитических отчетов по продажам и посещениям.

**3. Требования к системе**

3.1 **Функциональные требования**

* Бронирование услуг: Возможность выбора мастера, услуги, времени и даты.
* Управление расписанием: Добавление, редактирование и удаление записей в календаре.
* Уведомления: Система напоминаний по SMS и email.
* Отзывы и рейтинг: Оставление и просмотр отзывов о мастерах.
* Отчеты: Статистика по количеству записей, популярным услугам и мастерам.

3.2 **Нефункциональные требования**

* Производительность: Время отклика интерфейса не более 2 секунд при стандартной нагрузке.
* Масштабируемость: Возможность работы с 100+ мастерами и 1000+ активных клиентов.
* Безопасность: Поддержка HTTPS, защита данных с использованием SSL и хэширование паролей.

**4. Требования к пользовательскому интерфейсу**

4.1 **Основные экраны**

**• Главная страница:**

Список услуг с фильтрацией по категории и мастеру.

**• Личный кабинет клиента:**

История записей, оплата, отзывы и настройка профиля.

**• Личный кабинет мастера:**

Управление расписанием и просмотр записей.

**• Панель администратора:**

Управление расписанием, учетными записями и отчетами.

4.2 **Навигация**

Система должна включать меню с разделами: «Главная», «Услуги», «Мастера», «Личный кабинет», «Контакты».

4.3 **Юзабилити**

* Простой и понятный интерфейс.

**5. Требования к технической реализации**

5.1 **Языки и технологии**

* **Backend**: Laravel.
* **Frontend**: HTML CSS.
* **База данных**: MySQL.

5.2 **Архитектура системы**

* REST API для взаимодействия между клиентом и сервером.
* Модульная архитектура с разделением логики.
* Хранение данных в реляционной базе данных.
  1. **Интеграции**
* Внешние уведомления: интеграция с сервисом email-уведомлений (например, SendGrid).

**6. Требования к безопасности**

6.1 **Аутентификация и авторизация**

* **Авторизация**: разграничение прав доступа — администраторы, разработчики, покупатели.

6.2 **Шифрование данных**

* **HTTPS** для всех передаваемых данных.
* **Хранение паролей**: хэширование через bcrypt.

6.3 **Политики доступа**

* Защита от SQL-инъекций и XSS.
* Ограничение по IP для административного доступа.

**7. Ограничения и допущения**

7.1 **Технические ограничения**

* NaN.

7.2 **Финансовые ограничения**

* Бюджет проекта: до 150000 нервных клеток, включая тестирование и документацию.

7.3 **Сроки выполнения**

* Полная реализация проекта — 1 месяц с начала разработки.

**8. Требования к тестированию и приемке**

8.1 **Типы тестирования**

* **Функциональное тестирование**: проверка всех реализованных функций.
* **Тестирование безопасности**: проверка на уязвимости.

8.2 **Критерии приемки**

* Полное выполнение всех функциональных и нефункциональных требований.

**9. Требования к документации**

9.1 **Пользовательская документация**

* Руководство пользователя с инструкциями по созданию, редактированию и управлению страница с игровым контентом.

9.2 **Техническая документация**

* Описание архитектуры, структура базы данных, API.

**10. План реализации**

10.1 **Этапы разработки**

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

10.2 **Сроки выполнения этапов**

* Дата начала: X X X года.
* Дата завершения: X X X года.

10.3 **Ответственные лица**

* Менеджер проекта: Выропаева Л.В.
* Ведущий разработчик: Выропаева Л.В.

**11. Приложения**

* **Приложение A**: Детализированные схемы архитектуры.
* **Приложение B**: Примеры интерфейсов.
* **Приложение C**: Технические спецификации для интеграции с API других систем.