**Техническое задание на разработку приложения для ресторана**

1. Введение

Приложение предназначено для упрощения взаимодействия между рестораном и клиентами. Оно должно обеспечивать возможность онлайн-заказа, просмотра меню, управления бронированиями и получения информации о специальных предложениях, подсказывать самые популярные позиции по меню.

2. Цели и задачи

* Обеспечить пользователям удобный интерфейс для заказа еды и напитков.
* Позволить пользователям бронировать столики.
* Информировать клиентов о специальных предложениях и акциях.
* Собрать отзывы и оценки клиентов о блюдах и обслуживании.
* Показывать концепцию места
* Для начальства показывать оборот, остаток на складе и срок годности товаров
* Показывать для руководства Кол-во часов сотрудников
* Кол-во налички, карты и qr кода
* Показывать камеры видеонаблюдения в живом времени
* Кол-во сотрудника и провизии
* Кол-во аппаратуры

3. Функциональные требования

3.1. Пользовательский интерфейс

* Регистрация и авторизация пользователей (через email, социальные сети).
* Просмотр меню с возможностью фильтрации по категориям (закуски, основные блюда, десерты, напитки).
* Возможность добавления блюд в корзину и оформления заказа и предзаказа.
* Функция бронирования столиков с выбором даты и времени.
* Раздел с акциями и специальными предложениями.
* Возможность оставлять отзывы и оценки с чаевыми персоналу.

3.2. Административная панель

* Управление меню (добавление, редактирование, удаление блюд).
* Управление бронированиями (просмотр, подтверждение, отмена).
* Просмотр статистики заказов и отзывов.
* Настройка акций и специальных предложений.
* Показывание состояния продукции (срок годности, кол-во и чего остается мало)

4. Нефункциональные требования

* Приложение должно быть доступно на платформах iOS и Android.
* Интерфейс должен быть интуитивно понятным и адаптивным.
* Обеспечение безопасности данных пользователей (шифрование, защита от SQL-инъекций).
* Высокая производительность и быстрая загрузка страниц.

5. Технологический стек

* Frontend: React Native или Flutter.
* Backend: Node.js с Express или Python с Django.
* База данных: PostgreSQL или MongoDB.
* Хостинг: AWS или Heroku.

6. Сроки реализации

* Этап 1: Анализ требований и проектирование (2 недели).
* Этап 2: Разработка пользовательского интерфейса (4 недели).
* Этап 3: Разработка серверной части и интеграция (5 недели).
* Этап 4: Тестирование и исправление ошибок (3 недели).
* Этап 5: Запуск и поддержка (неопределенный срок).

7. Бюджет

* Оценка затрат на разработку, тестирование и запуск приложения.

8. Заключение

Для легкой новигации клиента. Помощь управляющим в отслеживании внутренней и внешней ситуации. Экономия времени клиента. Легкость использования. Бонусная программа с акциями поможет привлечь клиентов и напоминать об этом месте.