**Техническое задание на разработку системы автоматизации доставки для Delivery Club**

**1. Общие сведения**

**1.1. Наименование проекта**  
Система автоматизации процессов доставки для сервиса Delivery Club.

**1.2. Заказчик**  
Компания Delivery Club.

**1.3. Разработчик**  
[Название вашей компании или подразделения].

**1.4. Основание для разработки**  
Необходимость оптимизации процессов обработки заказов, повышения эффективности работы курьеров и улучшения качества обслуживания клиентов.

**1.5. Назначение системы**  
Автоматизация ключевых процессов, включая управление заказами, маршрутизацию курьеров, обработку платежей и формирование отчетности.

**2. Требования к системе**

**2.1. Функциональные требования**

* **Прием и обработка заказов:**
  + Автоматическое создание заказа с интеграцией меню ресторанов.
  + Поддержка различных способов оплаты.
  + Учет промокодов и скидок.
* **Управление курьерами:**
  + Автоматическое распределение заказов с учетом локации и загрузки курьеров.
  + Трекинг местоположения курьеров в реальном времени.
* **Обработка платежей:**
  + Интеграция с платежными системами.
  + Поддержка возвратов и отмены заказов.
* **Аналитика и отчетность:**
  + Формирование отчетов по заказам, курьерам и ресторанам.
  + Визуализация данных для анализа эффективности.

**2.2. Нефункциональные требования**

* **Производительность:**
  + Система должна обрабатывать до 1,000 заказов в минуту.
  + Время отклика API не должно превышать 500 мс.
* **Безопасность:**
  + Использование шифрования данных (SSL/TLS).
  + Аутентификация и авторизация пользователей.
* **Масштабируемость:**
  + Возможность увеличения нагрузки на 30% ежегодно.
* **Совместимость:**
  + Поддержка мобильных платформ iOS и Android.
  + Интеграция с ERP-системами ресторанов.

**3. Описание модулей системы**

**3.1. Модуль заказов**

* Создание, редактирование и отмена заказов.
* Расчет стоимости с учетом скидок и промокодов.
* Управление статусами заказов: "Новый", "В обработке", "Передан курьеру", "Доставлен".

**3.2. Модуль курьеров**

* Назначение заказов с учетом локации и загруженности курьеров.
* Трекинг местоположения и статуса курьеров.
* Учет рабочего времени и рейтинга курьеров.

**3.3. Модуль аналитики**

* Дашборды для анализа эффективности работы курьеров и ресторанов.
* Отчеты по ключевым метрикам: среднее время доставки, конверсия заказов.

**4. Технологии и инструменты**

* **Бэкенд:** Python, Django REST Framework.
* **Фронтенд:** React.js (для административной панели).
* **Базы данных:** PostgreSQL (основная), Redis (кеширование).
* **Инфраструктура:** Docker, AWS EC2.
* **Документация API:** Swagger.

**5. Этапы реализации**

1. **Анализ и проектирование** (3 недели):
   * Сбор требований.
   * Прототипирование интерфейсов.
2. **Разработка** (10 недель):
   * Реализация модуля заказов.
   * Интеграция с картографическими сервисами.
   * Настройка платежных шлюзов.
3. **Тестирование** (4 недели):
   * Функциональное и нагрузочное тестирование.
   * Исправление ошибок.
4. **Внедрение** (3 недели):
   * Поэтапный запуск системы.
   * Обучение пользователей.

**6. Критерии приемки**

* Полное соответствие функциональным требованиям.
* Успешное прохождение нагрузочного тестирования.
* Сокращение времени доставки на 15%.