Техническое задание на разработку мобильного приложения для прослушивания музыки “ TuneWave”

**1. Общие сведения**

* **Название проекта:** Мобильное приложение для прослушивание музыки “ Owoston”
* **Цель проекта:** Создание удобного и функционального мобильного приложения для пользователей, позволяющего наслаждаться музыкой в любое время и в любом месте, с возможностью создания персональных плейлистов, прослушивания радио, поиска новых исполнителей и треков.
* **Платформы**: iOS, Android.
* **Целевая аудитория:** Любители музыки всех возрастов, пользователи смартфонов и планшетов.

**2. Функциональные требования**

***2.1. Главное меню***

* В главном меню пользователь должен видеть:

1. Каталог музыки:
2. Популярные треки и альбомы.
3. Рекомендации на основе прослушиваний.
4. Новые релизы.
5. Плейлисты:
6. Созданные пользователем.
7. Подборки от приложения.

3.Поиск:

1. По названию трека, альбома, исполнителя.
2. По жанрам, настроению, году выпуска.

4.Личный кабинет:

1. Профиль пользователя.
2. Настройки.
3. История прослушивания.

***2.2. Каталог музыки***

* Для каждого трека должны отображаться:

1. Обложка альбома.
2. Название трека.
3. Исполнитель.
4. Длительность.
5. Кнопка "Слушать".
6. Кнопка "Добавить в плейлист".

* Для каждого альбома должны отображаться:

1. Обложка альбома.
2. Название альбома.
3. Исполнитель.
4. Год выпуска.
5. Список треков.
6. Кнопка "Слушать весь альбом".

* Для каждого исполнителя должны отображаться:

1. Фотография или логотип.
2. Имя исполнителя.
3. Список альбомов.
4. Список популярных треков.
5. Кнопка "Подписаться".

***2.3. Плейлисты***

* Пользователь может:

1. Создавать новые плейлисты.
2. Добавлять треки в существующие плейлисты.
3. Удалять треки из плейлистов.
4. Переименовывать плейлисты.
5. Удалять плейлисты.
6. Делиться плейлистами с другими пользователями.

* Плейлисты могут быть:

1. Публичными (видимыми для всех пользователей).
2. Приватными (видимыми только для создателя).

***2.4. Поиск***

* Поиск должен поддерживать:

1. Полнотекстовый поиск по названию трека, альбома, исполнителя.
2. Фильтрацию по жанрам, настроению, году выпуска.
3. Результаты поиска должны быть отсортированы по релевантности.

***2.5. Личный кабинет***

* Функционал личного кабинета должен включать:

1. Регистрация/авторизация (email, телефон, социальные сети).
2. Просмотр истории прослушивания.
3. Просмотр созданных плейлистов.
4. Управление подписками на исполнителей.
5. Настройки уведомлений (о новых релизах, начале трансляций, спецпредложениях).
6. Возможность редактирования профиля (имя, номер телефона, email).

***2.6. Уведомления***

* Push-уведомления:

1. О новых релизах исполнителей, на которых пользователь подписан.
2. О начале трансляций радиостанций, на которые пользователь подписан.
3. О специальных предложениях и скидках.
4. Напоминания о предстоящих событиях (концерты, фестивали).

**3. Нефункциональные требования**

***3.1. Требования к интерфейсу***

* Приложение должно иметь интуитивно понятный интерфейс.
* Поддержка языков:

1. Русский,
2. Английский.

* Поддержка адаптивного дизайна для различных размеров экранов.

***3.2. Производительность***

* Приложение должно быть оптимизировано для быстрого отображения данных и отзывчивости.
* Время загрузки основных страниц — не более 2 секунд.

***3.3. Безопасность***

* Все персональные данные пользователей должны быть защищены в соответствии с актуальными стандартами безопасности (шифрование, защита платежных данных).

***3.4. Интеграции***

* Интеграция с популярными платежными системами (Apple Pay, Google Pay).
* Возможность интеграции с системами лояльности (например, бонусные баллы, скидки для постоянных клиентов).
* Интеграция с сервисами потокового аудио (Spotify, Apple Music).

***3.5. Поддержка и обновления***

* Возможность расширения функционала приложения в будущем (например, добавление новых сервисов или возможностей).

**4. Требования к дизайну**

* Современный и привлекательный дизайн в соответствии с последними тенденциями UX/UI.
* Простота и удобство использования.

**5. Технические требования**

***5.1. Backend***

* Серверная часть должна быть реализована на современных технологиях, поддерживающих высокую нагрузку и стабильность (например, Node.js, Python).
* База данных: MySQL, PostgreSQL, MongoDB или аналогичная.

***5.2. Frontend***

* Мобильное приложение должно быть нативным для платформ iOS и Android.
* Используемые технологии: Swift (iOS), Kotlin/Java (Android).

**6. Тестирование**

* Проведение функционального и нагрузочного тестирования.
* Тестирование на различных устройствах с разными версиями ОС (Android, iOS).

**7. Дополнительные требования**

* Возможность интеграции с системами аналитики (Google Analytics, Firebase).
* Поддержка аналитики для отслеживания поведения пользователей.

**8. Сроки и этапы разработки**

* Этапы разработки:
* Анализ требований и проектирование: 2 недели.
* Разработка backend: 4 недели.
* Разработка frontend (iOS): 6 недель.
* Разработка frontend (Android): 6 недель.
* Тестирование и отладка: 3 недели.
* Запуск и поддержка: 2 недели.
* Общий срок разработки: 23 недели.

***9. Бюджет***

* Бюджет проекта согласовывается отдельно.