**Part 1: Выбор Сценария**

Для данной работы выбран сценарий: Сервис потоковой передачи музыки. Эта система будет обеспечивать удобство, персонализацию и масштабируемость в сфере конкретного музыкального приложения.

**Part 2: Проектирование Базы Данных и Документация**

**Идентификация Сущностей и Атрибутов:**

**1 Пользователи (Users)**

**2 Музыкальные треки (Tracks)**

**3 Типы подписок (Subscriptions)**

**4 Пользовательские плейлисты (Playlists)**

**Проектирование Таблиц:**

1 Table Name: Users

**Description:** Хранит информацию о пользователях

**Attributes:** User ID: INTEGER, PK, NOT NULL, UNIQUE

Username: VARCHAR(100), NOT NULL

Email: VARCHAR(250), UNIQUE, NOT NULL

Password: BIGINT, NOT NULL, UNIQUE

Country: VARCHAR(100)

**Constraints:**

PK\_ User: PRIMARY KEY (User ID)

UQ\_ Email: UNIQUE

UQ\_ Password: UNIQUE

2 Table Name: Tracks

**Description:** Содержит информацию о треках

**Attributes:** Track ID: INTEGER, PK, NOT NULL

Title: VARCHAR(100), NOT NULL

Album ID: INTEGER, NOT NULL

Artist ID: INTEGER, NOT NULL, UNIQUE

Genre: VARCHAR(100), NOT NULL

User ID: INTEGER, FK, NOT NULL, UNIQUE

Playlist ID: INTEGER, FK, NOT NULL

**Constraints:**

PK\_Track: PRIMARY KEY (Track ID)

FK\_User ID: FOREIGN KEY ( User ID) REFERENCES users\_table (User ID)

FK\_Playlist ID: FOREIGN KEY ( Playlist ID) REFERENCES playlists\_table (Playlist ID)

3 Table Name: Subscriptions

**Description:** Хранит информацию о типах подписок

**Attributes:** Subscription ID: INTEGER, PK, NOT NULL, UNIQUE

Name: VARCHAR(100), NOT NULL

Price: NUMERIC, NOT NULL

User ID: INTEGER, FK, NOT NULL, UNIQUE

**Constraints:**

PK\_Subsciption: PRIMARY KEY (Subscription ID)

CHK\_Price: CHECK (Price >= 0$)

FK\_User ID: FOREIGN KEY ( User ID) REFERENCES users\_table (User ID)

4 Table Name: Playlists

**Description:** Хранит информацию о пользовательских плейлистах

**Attributes:** Playlist ID: INTEGER, PK, NOT NULL, UNIQUE

User ID: INTEGER, FK, NOT NULL

Title: VARCHAR(100), NOT NULL

Track ID: INTEGER, FK, NOT NULL

**Constraints:**

PK\_Playlist: PRIMARY KEY (Playlist ID)

FK\_User ID: FOREIGN KEY (User ID) REFERENCES users\_table(User ID)

FK\_Track ID: FOREIGN KEY (Track ID) REFERENCES tracks\_table(Track ID)

**Взаимосвязи:**

* **Users и Tracks** (один-ко-многим). Один пользователь может слушать множество различных треков. Как и у одного трека, есть множество слушателей.

Track. User ID является внешним ключом ссылающим на Users. Useres ID

* **Users и Subscriptions** (один-ко-многим). Один пользователь может иметь несколько подписок.

Subscriptions. User ID является внешним ключом ссылающим на Users. Users ID

**Tracks и Playlists** (многие-ко-многим). Один плейлист может содержать в себе множество треков. Так же как один трек может входить во множество плейлистов.

Track. Users ID является внешним ключом ссылающим на Users. Users ID

Track. Playlist ID является внешним ключом ссылающим на Playlists. Playlist ID

**Playlists и Users** (один-ко-многим). Пользователь один, а плейлистов у него может быть много.

Playlists. User ID является внешним ключом ссылающим на Users. User ID

Playlists. Track ID является внешним ключом ссылающим на Track. Track ID

**Artist и Tracks** (один-ко-многим). Исполнитель один, а песен у него несколько.

Artist. Track ID является внешним ключом ссылающим на Artist. Artist ID

**Album и Tracks.** (один-ко-многим). Альбом один, а треков в нем несколько.

Album. Track ID является внешним ключом ссылающим на Tracks. Track ID

**Artist и Album** (один-ко-многим). У одного исполнителя может быть несколько альбомов.

Artist. Album ID является внешним ключом ссылающим на Album. Album ID