**Symphony Project Design Document**

**Client Information:**

* **Client**: Works for YouTube (contract through a third party).
* **Industry**: Music business.
* **Requirement**: Integrate YouTube's music algorithm into a project.

**Project Overview:**

**Phase 1: API Access**

* **Spotify API**: Reference for API call structure (setup complete).
* **YouTube Advanced API**: Need access via client's connections.
* **YouTube Basic API**: If sufficient, implement using .get in Python.

**Phase 2: Algorithm and Features**

* **Objective**: Develop an algorithm for music suggestions based on user emotions.
  + Suggestions for three emotions: Happy, Sad, Excited.
  + Use a scale of 0-10 to quantify and make music suggestions.
* **Web Interface**:
  + Scalable design using React and server-based studies.
  + Potential feature: Voice pitch analysis to detect user emotion.
* **Database**: Create a relational database for storing and accessing data.

**Technical Requirements:**

**Programming Languages:**

* **Backend**: Python.
* **Frontend**: JavaScript (React), HTML, CSS.

**Database Options:**

* **SQLite**: Lightweight option.
* **JSON**: Suitable for testing phase.
* **SQL**: Preferred for production backend.

**Hosting Options:**

* **Replit**: For initial testing (like CodePen).
* **AWS Free Tier**: 12-month free usage (client funding required after free period).
* **Docker**: Research and consider for deployment.

**Development Environment:**

* **IDE**: Visual Studio.

**Implementation Details:**

**Client Information and Privacy:**

* **NDA**: Client NDA with YouTube cleared on 07/22/2024.
* **Privacy**: All client information to be anonymized for public streaming.

**Streaming and Testing Platforms:**

* **Platforms**: Twitch, GitHub, YouTube, Facebook, TikTok, and other approved Coding with Strangers platforms.
* **Testing**: Use free web hosting sites to gather audience feedback.

**Execution Plan:**

1. **Setup Development Environment**:
   * Configure Visual Studio for Python and React development.
   * Set up initial database structure using SQLite or JSON for testing.
2. **API Integration**:
   * Integrate Spotify API for reference.
   * Obtain access to YouTube Advanced API and integrate.
   * Develop functions to pull data using YouTube Basic API if Advanced API is not available.
3. **Algorithm Development**:
   * Design and implement the music suggestion algorithm.
   * Define parameters for happy, sad, and excited music suggestions.
4. **Web Interface**:
   * Develop a scalable web interface using React.
   * Implement voice pitch analysis feature (optional).
5. **Database and Hosting**:
   * Finalize database structure using SQL for production.
   * Choose appropriate hosting solution and deploy.
6. **Testing and Deployment**:
   * Conduct initial tests using Replit or other free hosting platforms.
   * Gather user feedback and make necessary adjustments.
   * Deploy final version on chosen hosting solution.