**Requirement Gathering and Analysis Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date |  |
| Team ID |  |
| Project Name | StreamSavvy |
| Maximum Marks |  |

**Solution Architecture for OTT Platform**

**Overview:** The solution architecture serves as a comprehensive plan bridging the business needs and technological implementation. Its goal is to identify the best technology stack and approach to solve existing business challenges, outline structural and behavioral elements, and set clear development phases and solution requirements.

**Solution Details:**

**1. Business Problem and Objectives:**

* **Business Problem:** The need for a seamless OTT platform capable of handling multiple user requests efficiently, providing a high-quality user experience, and integrating with multiple data sources for diverse media content.
* **Objective:** To build an efficient, scalable, and secure web application capable of supporting high user loads and providing real-time content delivery with minimal latency.

**2. Solution Structure and Components:**

* **Frontend Application:**
  + **UI Module**: Handles user interaction, navigation, and display.
  + **Data Display Module**: Manages how the media content is rendered and customized based on user preferences.
* **Backend Service (API Service)**:
  + **Data Endpoint**: Receives data requests, processes them, and sends responses back to the frontend. Integrates with media databases and third-party APIs for media content and metadata.
  + **Business Logic**: Processes user authentication, content recommendations, and other key functionalities.

**3. Development Phases:**

* **Phase 1**: Requirement Gathering and Analysis
* **Phase 2**: Architecture Design (Current Phase)
* **Phase 3**: Frontend and Backend Development
* **Phase 4**: Testing and Optimization
* **Phase 5**: Deployment and Monitoring

**4. Solution Requirements:**

* **Functional Requirements**:
  + User Authentication
  + Content Search and Display
  + Real-time Video Playback
* **Non-Functional Requirements**:
  + High availability and scalability
  + Secure data handling and storage

**5. Specifications:**

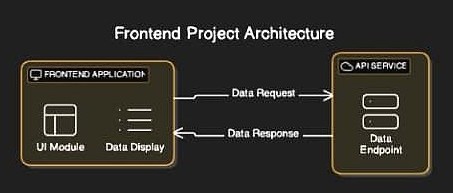
* **UI/UX Framework**: React.js for responsive and dynamic user interface.
* **API Management**: Node.js/Express for building RESTful APIs.
* **Database**: MongoDB or a similar NoSQL solution for handling large-scale media content.
* **Security Measures**: Implementation of OAuth 2.0, data encryption using SSL/TLS, and other access control measures.
* **Performance Enhancements**: Load balancing, use of CDNs, and efficient data caching mechanisms.

**Solution Architecture:**

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

* Find the best tech solution to solve existing business problems.
* Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
* Define features, development phases, and solution requirements.
* Provide specifications according to which the solution is defined, managed, and delivered.

**Example - Solution Architecture Diagram:**

**s**