**Software Design Document (SDD) for UniVoice**

**1. Introduction**

**1.1 Purpose**

UniVoice a College News Management System that integrates news publication and radio streaming. It covers the architectural design, module descriptions, and external interfaces necessary for implementation.

**1.2 Scope**

The system provides role-based access to users, allowing them to post and manage news articles, listen to live or recorded radio streams, and facilitate real-time communication. The design ensures scalability, security, and usability.

**2. System Overview**

**2.1. System functions**

UniVoice is a web-based platform that replaces traditional notice boards with digital news management and radio broadcasting. The system includes functionalities for:

* News posting, categorization, and approval.
* Role-based access control (Admin, Faculty, Students).
* Multimedia support (Images, Videos, PDFs).
* Live and recorded radio streaming.
* Search and filtering capabilities.

2.2. Constraints

* The system must be able to handle a large number of concurrent users.
* The system must be secure to prevent unauthorized access to user information and posting news.
* The system must be scalable to handle an increasing number of visitors and news.
* The system must be available 24/7.

**3. System Architecture**

**3.1 Architectural Design**

UniVoice follows a **three-tier architecture**:

* **Presentation Layer:** Web-based UI built using HTML, CSS, JavaScript, and React.
* **Application Layer:** Backend services implemented with Node.js and Express.js.
* **Data Layer:** Database management using MongoDB.

**3.2 System Components**

* **Frontend UI:** Responsive web interface.
* **Backend API:** Handles authentication, news management, and radio streaming.
* **Database:** Stores user data, news content, and archived broadcasts.
* **Streaming Server:** Integrated third-party audio streaming service.

3.3. Data architecture : UniVoice will use a NoSQL model for its database to manage the varying data . The Database will store user data , News information , User roles etc.

**4. Module Design**

**4.1 User Management Module**

* **Admin:** Manage users, approve news articles, oversee radio streaming.
* **Users :** Read and Post news, listen to radio streams.

**4.2 News Management Module**

* **Create, edit, delete news articles.**
* **Categorization into Academic, Events, Sports, Notices.**
* **Approval workflow for news publication.**

**4.3 Radio Streaming Module**

* **Live streaming support for college announcements.**
* **Recorded broadcasts stored for playback.**
* **Integration with third-party streaming services.**

**4.4 Search and Filter Module**

* **Keyword-based search.**
* **Filter by category, date, and author.**

**5. External Interfaces**

**5.1 User Interfaces**

* Web-based dashboard with admin panel.
* Mobile-friendly responsive design.

**5.2 Hardware Interfaces**

* Compatible with desktops, tablets, and smartphones.
* Requires web server and database server.

**5.3 Software Interfaces**

* Authentication: College authentication system (LDAP/SSO).
* Database: MongoDB.
* Streaming: Integrated with third-party audio streaming.

**6. Security and Performance Considerations**

**6.1 Security Measures**

* **Encrypted user data and secure authentication.**
* **Token-based sessions using JWT for secure access control.**
* **Data encryption for audio streaming.**

**6.2 Performance Considerations**

* **Load balancing for handling concurrent users.**
* **Efficient caching mechanisms.**