# Detailed Design Document: ChatSphere

## 1. Introduction

### 1.1 Purpose

This document aims to present a comprehensive overview of ChatSphere's design and architecture, encompassing features, functionality, and technical specifications.

### 1.2 Scope

ChatSphere, as a chatting app, endeavors to provide users with a dynamic platform for real-time communication, fostering seamless conversations, multimedia sharing, and interactive features to enhance the overall chatting experience.

### 1.3 Objectives

* Make it easy for users to chat by creating a simple and friendly interface in ChatSphere.
* Keep user information safe with strong security measures.
* Allow ChatSphere to grow and handle more users in the future.
* Let users work together by making it easy to share and collaborate on chats and activities..

## 2. System Overview

### 2.1 System Architecture

ChatSphere employs a client-server architecture with a responsive web-based front end for user interactions and a robust back-end server handling core logic, ensuring an efficient and dynamic chatting experience.

### 2.2 Key Features

* User registration and authentication
* Real-Time Chatting
* Notification System
* Video Calling
* Notification System
* Emoticons and Stickers

### 2.4 Technologies Used

* Front-end: React
* Back-end: Express
* Database: MongoDB
* Authentication: JWT
* Additional tools/libraries as needed.

## 3. Database Design

3.1 Entity-Relationship Diagram

User Entity:

* *UserID (Primary Key):* Unique identifier.
* *Username:* User's chosen name.
* *Email:* User's contact address.
* *Password:* Secure authentication.

Message:

* *Id (Primary Key):* Unique identifier..
* *SenderID:* Sender's UserID.
* *Data:* Message content.
* *SenderID:* Sender's UserID.

Friends:

* *Id(Primary Key):* Unique identifier.
* *UserID (Foreign Key):* Owner of the contact.
* *FriendId(Foreign Key):* UserID of the contact.

## 4. Database Diagram

## 

## 

## 

## 

## 