Here's a summary of the components I've developed:

1. **Database Structure**:
   * tbl\_users - Stores user information like username, fullname, password, email, and status
   * tbl\_chats - Stores chat messages with sender, receiver, message content, and read status
2. **Core Files**:
   * config.php - Database connection and core utility functions
   * index.php - Main entry point that redirects to login or dashboard
   * .htaccess - Security configurations for the web server
3. **User Authentication**:
   * login.php - User login form and authentication
   * register.php - New user registration
   * logout.php - User logout and status update
4. **Chat Functionality**:
   * dashboard.php - Main chat interface with user list and chat area
   * ajax\_chat.php - Backend API for chat operations (sending/receiving messages)
   * js/chat.js - JavaScript for real-time chat interactions
   * css/style.css - Styling for the entire application including responsive design
   * profile.php - User profile management

**Key Features Implemented:**

1. **User Management**:
   * User registration and login
   * User profile with information update capabilities
   * Online/offline status tracking
2. **Chat System**:
   * Real-time messaging between users
   * Message read status tracking
   * Unread message counts
   * User search functionality
3. **Mobile Responsiveness**:
   * Sidebar toggles on mobile devices
   * Responsive layout for all screen sizes
   * Touch-friendly interface elements
4. **Security Features**:
   * Input sanitization to prevent SQL injection
   * Password hashing (commented placeholders in the code)
   * Session management
   * Protected sensitive files

To use the application:

1. Create a database called userchatdb2
2. Import the SQL code from the database structure artifact
3. Place all PHP files in your web directory
4. Create a css folder for the style.css file and a js folder for the chat.js file
5. Create an images folder with a default.jpg placeholder image
6. Access the application by visiting the index.php page

This implementation follows the requirements you specified, using the database structure with 'sender' and 'receiver' fields for tracking chat messages, and loading user details from the users table.