



## **INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT, AKURDI, PUNE**

### **“Connexus”**

**PG-DAC August 2025**

Submitted By:

Group No: 48

<b>Roll No.</b>	<b>Name of Student</b>
-----------------	------------------------

258058	Pranav Patil
--------	--------------

258176	Raj Dhakar
--------	------------

**Mrs.Neelam**  
Project Guide

**Mr. Anil Sharma**  
Centre Coordinator

## **ABSTRACT**

**Connexus** is a high-performance, scalable professional networking platform designed to facilitate seamless interaction, content sharing, and real-time communication. Leveraging a modern Microservices Architecture, the platform integrates complex networking logic with real-time social features to provide a robust user experience tailored for professional networking.

The platform's frontend is built using React and Material UI, ensuring a responsive, premium, and intuitive user interface with smooth animations powered by Framer Motion. It utilizes StompJS and WebSockets to provide real-time messaging and instant notifications.

On the backend, Connexus employs a suite of Spring Boot Microservices, including dedicated services for User Management, Posts, Connections, Messaging, and Notifications. Key architectural components include Netflix Eureka for service discovery and a centralized API Gateway for secure routing. Security is enhanced through Spring Security with JWT-based authentication, while media assets are managed and delivered via Cloudinary. By decoupling core functionalities into independent services, Connexus achieves high scalability and fault tolerance, making it a robust for professional communities.

## **ACKNOWLEDGEMENT**

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my heartfelt thanks to our esteemed guide, **Mrs.Neelam** for providing me with the right guidance and advice at the crucial juncture and showing me the right way. I sincerely thank our respected Centre Co-Ordinator, Mr.Anil sharma, for allowing us to use the available facilities. I would also like to thank the other faculty members at this occasion. Last but not least, I would like to thank my friends and family for the support and encouragement they have given me during our work.

Pranav Patil (240341220055)

Raj Dhakar (2403412200156)

## **Table of Contents**

Sr.No	Description	Page No.
1	Introduction	1
2	SRS	9
3	Diagrams	15
3.1	ER Diagram	15
3.2	Use Case Diagram	16
3.3	Data Flow Diagram	17
3.4	Activity Diagram	18
3.5	Class Diagram	21
3.6	Component Diagram	22
3.7	Sequence Diagram	23
4	Database Design	24
5	Snapshots	27
6	Conclusion	32
7	References	33

## **1. INTRODUCTION**

**Connexus** is a high-performance professional networking and social interaction platform designed to bridge the gap between career opportunities and professional community engagement. It is a comprehensive system built on a Microservices Architecture, integrating advanced networking logic with real-time social functionalities. Connexus allows professionals to showcase their expertise, share industry insights through multimedia posts, and build meaningful professional relationships through a secure and scalable environment.

The platform utilizes React and Material UI for a sophisticated, responsive frontend that prioritizes user experience with smooth animations via Framer Motion. The backend is powered by a robust suite of Spring Boot Microservices, ensuring high availability and fault tolerance. Security is a core pillar of the platform, implemented through Spring Security for fine-grained authentication and authorization, utilizing JWT for stateless session management. Real-time features, such as instant messaging and notifications, are facilitated by WebSockets (StompJS).

Connexus is tailored to meet the needs of the modern workforce, providing tools for both job seekers (Professionals) and talent scouts (Recruiters), while maintaining a secure, role-governed ecosystem managed by system Administrators

## 1.1 Roles and Access Control

In the Connexus system, distinct roles—Admin, Professional, and Recruiter—ensure that tasks are appropriately delegated and the platform remains secure.

### 1. Admin Role

The Admin role has comprehensive control over the platform's infrastructure and community standards. Admins are responsible for the health and integrity of the professional network.

#### ❖ Responsibilities:

- **User & Content Auditing:** Monitor platform activity, manage user reports, and moderate content to ensure professional standards.
- **Service Monitoring:** Oversee the status of the various microservices (via Eureka) and the API Gateway.
- **System Configuration:** Manage global settings, including security policies and third-party integrations (Cloudinary, etc.).
- **Reporting & Analytics:** Generate insights on user engagement, network growth, and platform performance.
- **Access Level:**

Full access to administrative dashboards and system logs.

Ability to manage user permissions and handle security-related overrides.

### 2. Professional (User) Role

The Professional role is the core of the community, focusing on networking, content creation, and career growth.

❖ **Responsibilities:**

- **Profile Management:** Create and update a professional digital identity, showcasing skills, experience, and education.
- **Networking:** Send and receive connection requests, build a professional network, and interact with peers.
- **Content Sharing:** Create and engage with posts (text and media), follow industry trends, and participate in professional discussions.
- **Real-time Communication:** Securely message connections and receive live notifications for network activity.

▪ **Access Level:**

Access to personalized feeds, messaging modules, and network management tools.

No access to system configurations or other users' private administrative data.

**Access Level:**

Full access to administrative dashboards and system logs.

Ability to manage user permissions and handle security-related overrides.

**2. Professional (User) Role**

The Professional role is the core of the community, focusing on networking, content creation, and career growth.

**Responsibilities:**

Profile Management: Create and update a professional digital identity, showcasing skills, experience, and education.

Networking: Send and receive connection requests, build a professional network, and interact with peers.

- **Content Sharing:** Create and engage with posts (text and media), follow industry trends, and participate in professional discussions.
- **Real-time Communication:** Securely message connections and receive live notifications for network activity.
- **Access Level:**

Access to personalized feeds, messaging modules, and network management tools.

No access to system configurations or other users' private administrative data.

#### ❖ **Recruiter Role**

The Recruiter role is a specialized user type focused on talent acquisition and candidate engagement.

- **Responsibilities:**

Talent Discovery: Access specialized dashboards to find and filter professional talent.

Engagement: Communicate with potential candidates and manage recruitment-related interactions.

Dashboard Management: Utilize the Recruiter Dashboard to track engagement and talent pipelines.

- **Access Level:**

Access to recruitment-specific tools and dashboards.

Permissions to view professional profiles and initiate professional outreach.

## 2. Professional (User) Role

The Professional role is the core of the community, focusing on networking, content creation, and career growth.

### **Responsibilities:**

**Profile Management:** Create and update a professional digital identity, showcasing skills, experience, and education.

**Networking:** Send and receive connection requests, build a professional network, and interact with peers.

**Content Sharing:** Create and engage with posts (text and media), follow industry trends, and participate in professional discussions.

**Real-time Communication:** Securely message connections and receive live notifications for network activity.

### **Access Level:**

Access to personalized feeds, messaging modules, and network management tools.

No access to system configurations or other users' private administrative data.

## 3. Recruiter Role

The Recruiter role is a specialized user type focused on talent acquisition and candidate engagement.

**Responsibilities:**

**Talent Discovery:** Access specialized dashboards to find and filter professional talent.

**Engagement:** Communicate with potential candidates and manage recruitment-related interactions.

**Dashboard Management:** Utilize the Recruiter Dashboard to track engagement and talent pipelines.

**Access Level:**

Access to recruitment-specific tools and dashboards.

Permissions to view professional profiles and initiate professional outreach.

**1.2 Purpose**

The purpose of Connexus is to provide a unified, modern environment where professionals can connect, communicate, and grow. By leveraging a microservices architecture, the system aims to solve the scalability issues of traditional monolithic social platforms while providing a highly responsive and secure user experience. It streamlines professional networking by integrating real-time messaging, media sharing, and talent acquisition into a single cohesive platform.

**1.3 Scope**

The scope of the Connexus project encompasses the full-stack development of a microservices-based social ecosystem. This includes:

Developing independent services for Users, Posts, Connections, Messaging, and Notifications.

Implementing a centralized API Gateway and Service Discovery (Eureka).

Creating a premium React-based frontend with a focus on professional aesthetics.

Integrating Cloudinary for scalable media storage and WebSockets for real-time delivery.

## 1.4 Objectives of Connexus

**Distributed Scalability:** Utilize microservices to allow independent scaling of high-traffic features like messaging and posts.

**Real-Time Engagement:** Provide sub-second latency for chats and notifications via persistent WebSocket connections.

**Robust Security:** Ensure data privacy and secure communication using JWT and Spring Security.

**Professional UI/UX:** Deliver a high-end interface that feels premium and promotes professional interaction.

**Seamless Media Integration:** Support multimedia post creation with efficient cloud storage.

**Reliable Networking Logic:** Implement complex connection algorithms to manage professional relationships effectively.

## 1.5 Functionalities Provided by Connexus

### 1. Identity & User Management

**Secure Authentication:** JWT-based login and registration with encrypted password storage.

**Granular Profiles:** Detailed professional profiles with media support via Cloudinary.

**Role-Based Access (RBAC):** Strict enforcement of Admin, Professional, and Recruiter permissions.

### 2. Content & Post Management

**Multimedia Sharing:** Create posts with text and images, managed via the post-service.

**Interactive Feeds:** Real-time social feed with cascading updates and interactive engagement.

**Activity Tracking:** Monitor "My posts" and network interactions through a dedicated UI.

## ❖ Professional Networking (Connection Service)

- **Invitation System:** Manage incoming and outgoing professional invitations.
- **Network Mapping:** Track professional relationships and suggestions.
- **Dynamic Actions:** Accept, ignore, or remove connections with real-time state updates.

## ❖ 4. Real-Time Messaging & Notifications

- **Instant Chat:** Peer-to-peer messaging using the messaging-service and WebSockets.
- **Live Notifications:** Real-time alerts for messages, connections, and post interactions.
- **Message Persistence:** Storing and retrieving chat history across sessions.

## ❖ 5. Specialized Recruiter Tools

- **Recruiter Dashboard:** Centralized hub for managing talent discovery and recruitment workflows.
- **Candidate Search:** Enhanced visibility into professional talent within the network.

## ❖ 6. Architectural Resilience

- **Service Discovery:** Automated registration and discovery of services via Eureka.
- **Gateway Routing:** Single entry point for all frontend requests, simplifying client-side API logic.
- **Fault Tolerance:** Decoupled services ensure that a failure in one module (e.g., Notifications) doesn't crash the entire platform.

## **2. SOFTWARE REQUIREMENT SPECIFICATION**

The functional requirements for Connexus define the specific features and technical capabilities that the system must provide to support a robust professional networking ecosystem. These requirements ensure that the microservices architecture effectively addresses user needs and business objectives.

### **2.1 Functional Requirements for Connexus**

#### **1. Identity and User Management**

- User Registration and Login:**

The system shall allow users to create professional accounts and authenticate securely using JWT-based login mechanisms

- Role-Based Access Control (RBAC):**

The system shall enforce distinct permissions for Professionals, Recruiters, and Admins to ensure data privacy and operational integrity.

- Profile Management:**

Users shall be able to build and update detailed professional profiles, including designations, skills, professional summaries, and profile images hosted via Cloudinary.

#### **2. Post and Content Management**

- Multi-Media Post Creation:**

The system shall allow users to publish posts containing titles, long-form descriptions, and media attachments (images/videos).

- Interactive Professional Feed:**

The system shall provide a dynamic feed of posts from the user's network, supporting real-time interaction (likes, comments)

- **Personal Activity Tracking:**

Users shall have a dedicated "My Posts" section to view and manage their historical content..

### **3. Professional Networking (Connection Service)**

- **Connection Requests:**

Users shall be able to send, receive, and manage professional connection requests to expand their network

- **Invitation Management:**

The system shall provide a centralized interface for users to "Accept" or "Ignore" incoming invitations.

- **Network Visibility:**

The system shall allow users to view their current professional network and discover potential connections.

### **4. Real-Time Messaging and Notifications**

- **Instant Peer-to-Peer Chat:**

The system shall support real-time messaging between connections using WebSockets (StompJS).

- **Live Notifications:**

The system shall deliver instant alerts for new connection requests, messages, and social interactions.

- **Message History:**

The system shall persist chat history, allowing users to retrieve previous conversations across sessions.

## 5. Specialized Recruiter Dashboard

- **Talent Search & Discovery:**

Recruiters shall have access to advanced search tools to find professional talent based on skills, location, and experience.

- **Recruitment Management:**

Recruiters shall have access to advanced search tools to find professional talent based on skills, location, and experience.

## 6. Security & Infrastructure

- **Stateless Authentication:**

The system shall use JWT for secure, stateless communication between the frontend and microservices.

- **Service Discovery & Routing:**

The system shall utilize Netflix Eureka for service discovery and an API Gateway for centralized request management and security.

## 2.2 Non-Functional Requirements for Inventory SellMart

### 1. Performance

- **Response Time:**

The system shall respond to UI Interaction within 2 seconds under normal operating conditions.

- **Scalability:**

The microservices architecture shall support horizontal scaling to handle increasing user loads and data volume.

- **Real-time Latency:**

Messaging and notification delivery shall maintain sub-second latency via persistent WebSocket connections.

### 2. Reliability & Availability

- **Premium UI Design:**

The interface shall be built with Material UI to provide a clean, professional, and intuitive user experience.

- **Responsiveness:**

The platform shall be fully responsive, ensuring seamless operation across desktops, tablets, and smartphones.

- **Error Handling:**

The system shall gracefully handle errors and provide meaningful feedback to users when issues occur.

### 3. Usability

- **User Interface:**

The system shall have a user-friendly interface that is easy to navigate, with clear instructions and minimal learning curve.

### 4. Maintainability

- **Modularity:**

The system shall be designed in a modular fashion, allowing for easy updates and enhancements to individual components without affecting the entire system.

- **Other Requirements:**

**Hardware and Network Interfaces:****Back-end Server Configuration:**

- Processor: Multi-core (Intel i5/i7 or equivalent)
- RAM: 16 GB (Recommended for running multiple microservices)

**Client Configuration:**

- Processor: Modern dual-core/quad-core
- RAM: 8 GB+
- Display: 1080p resolution for optimal UI experience

**Software Interfaces:****Software configuration for back-end Services:**

- Java 17+
- Spring Boot (Auth, JPA, Gateway, Eureka)
- MySQL (User & Post data)
- Neo4j (Graph-based Connection data)

**Software configuration for front-end Services:**

- ReactJs
- Material UI
- Axios
- StompJs (WebSocket Client)

**External Services:**

- Cloudinary (media Storage API)
- JWT (Security Token Standard)
- StompJs (WebSocket Client)

### 3. DIAGRAMS

#### 3.1 Entity Relationship Diagram:

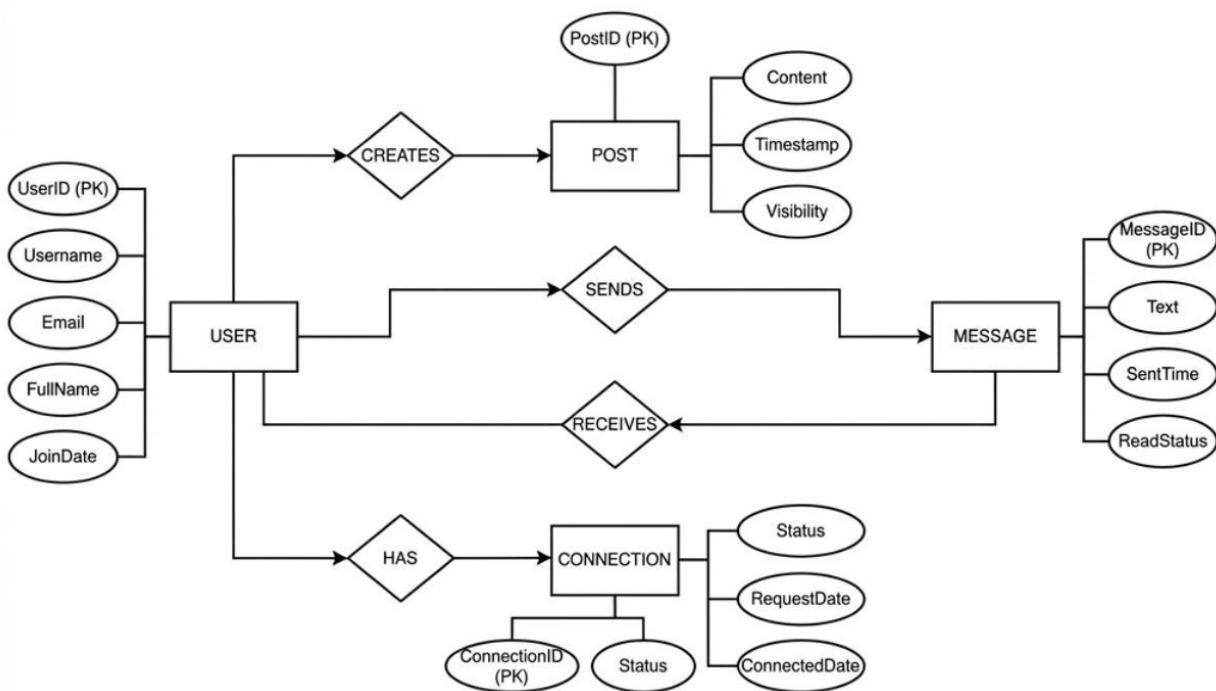


Fig. ER Diagram for Connexus

### 3.2 Use Case Diagram:

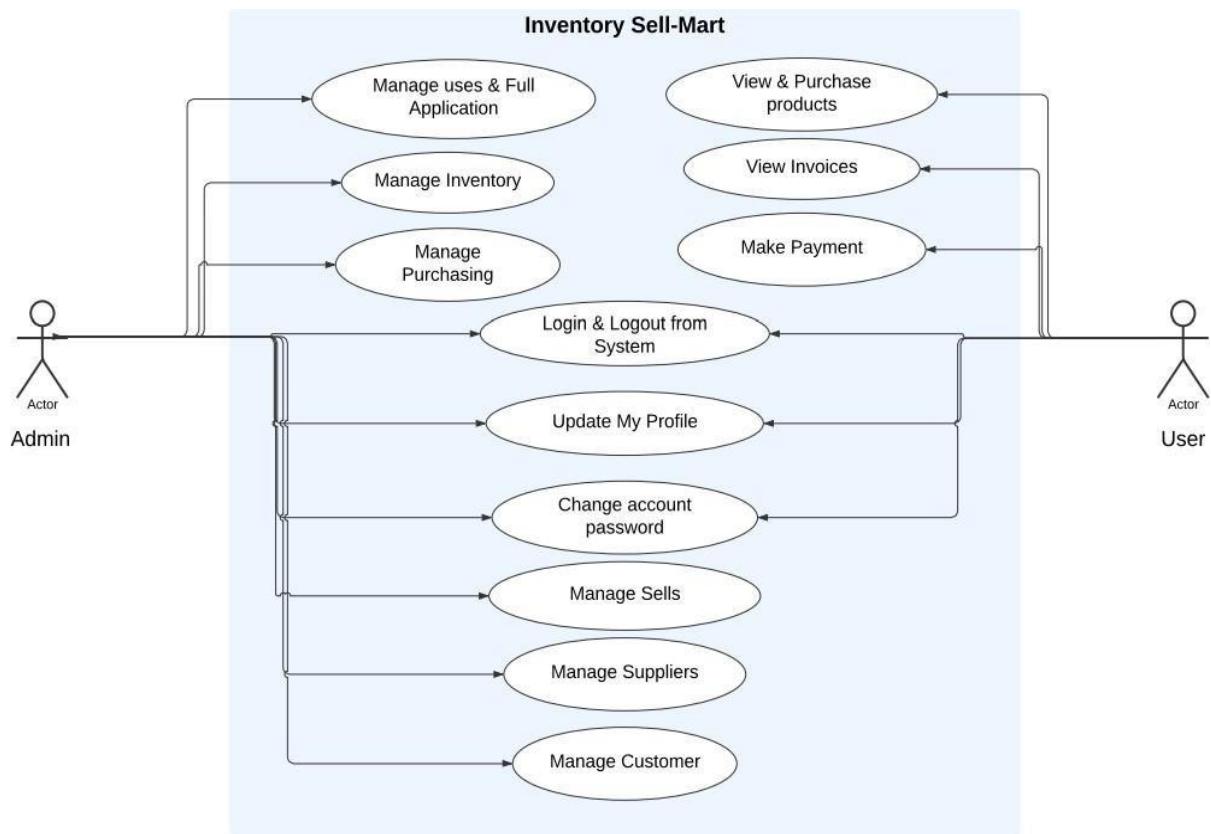
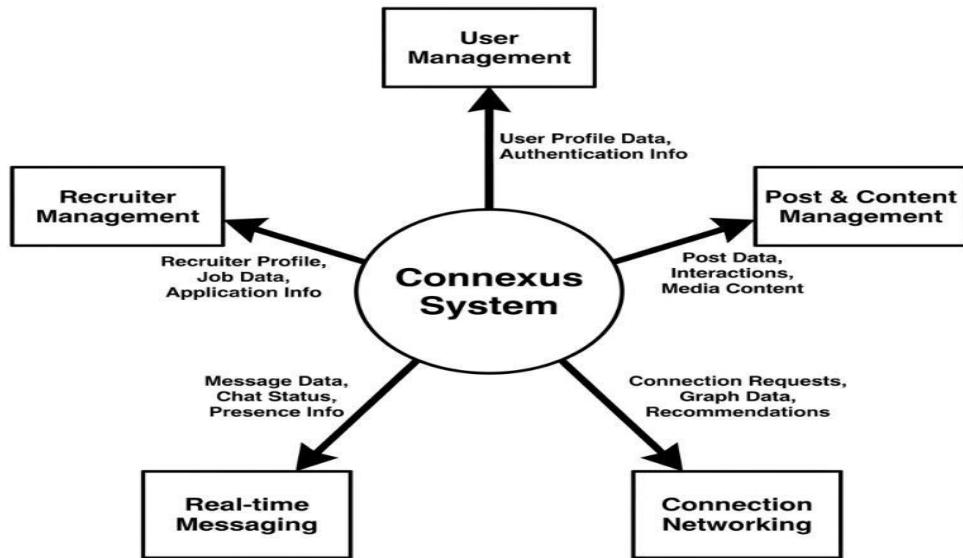


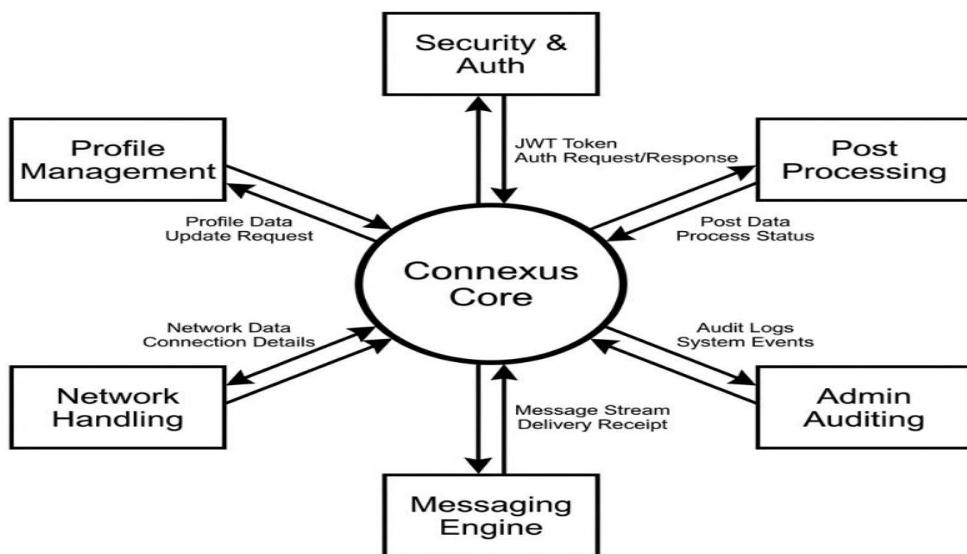
Fig. Use Case Diagram for Connexus

### 3.3 Data Flow Diagram:

#### DFD Level 0:



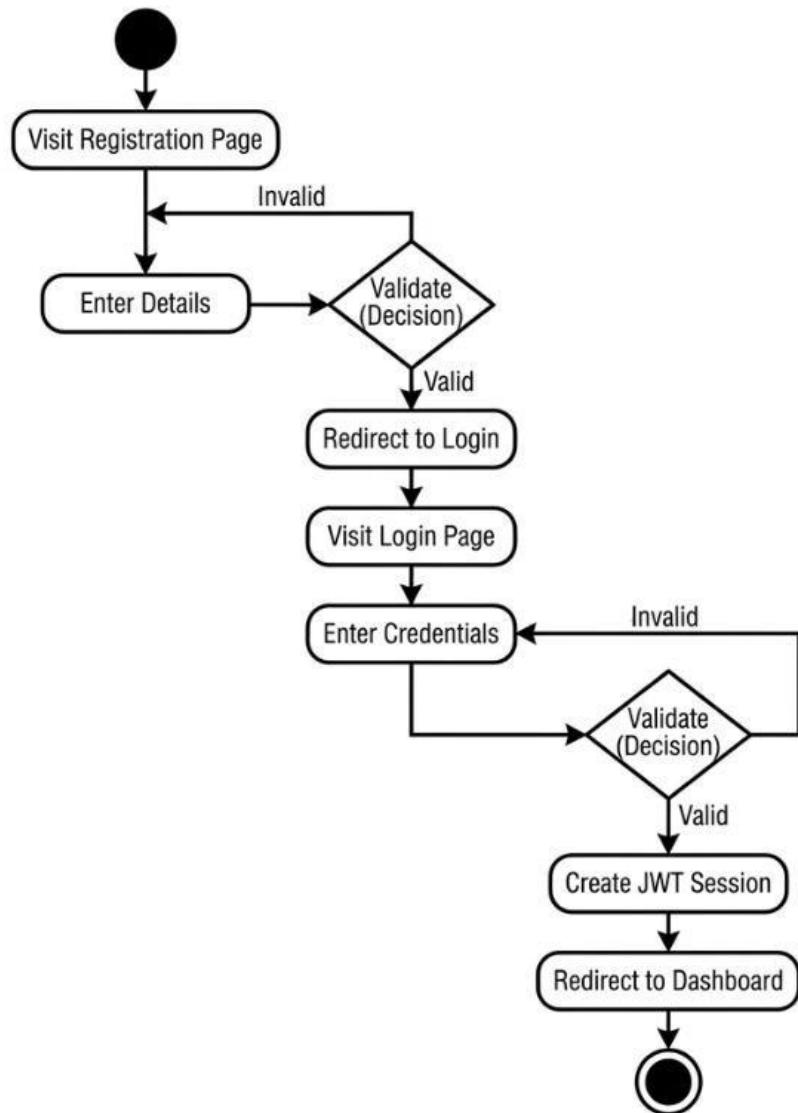
#### DFD level 1



### 3.4 Activity Diagram :

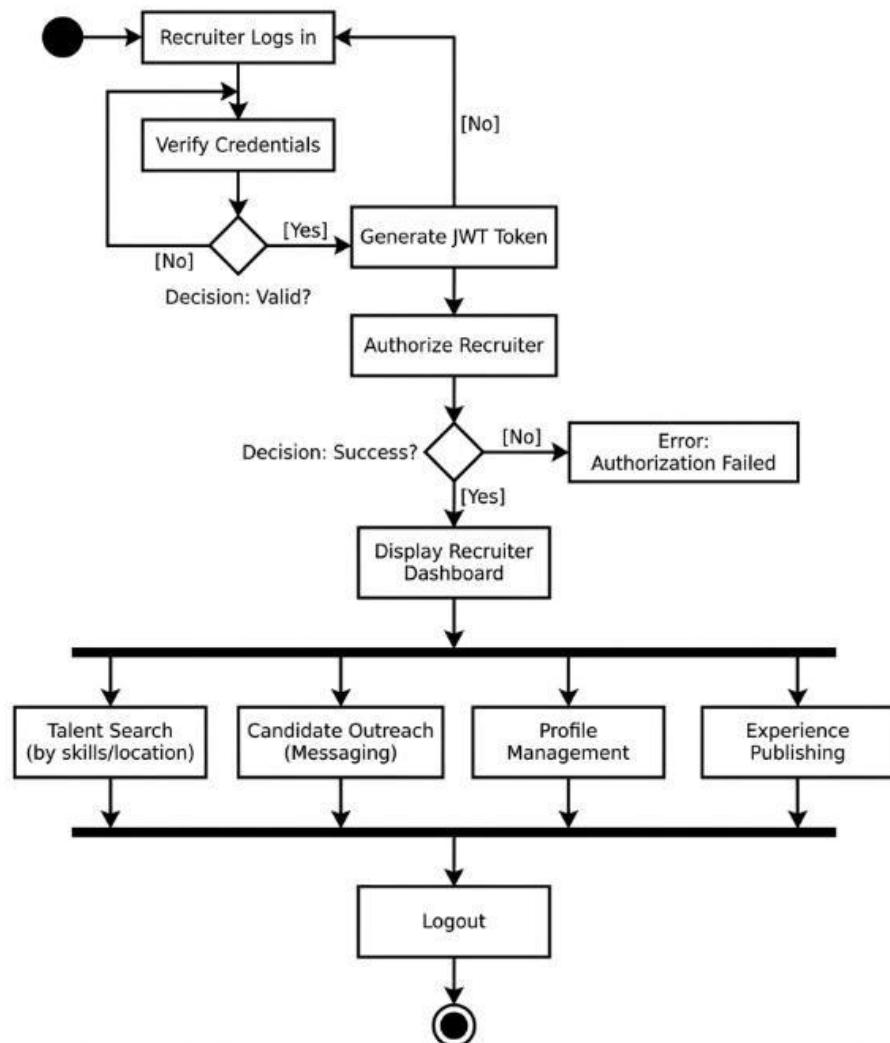
#### 1. Login Activity Diagram

### Connexus: Login and Registration Activity Diagram



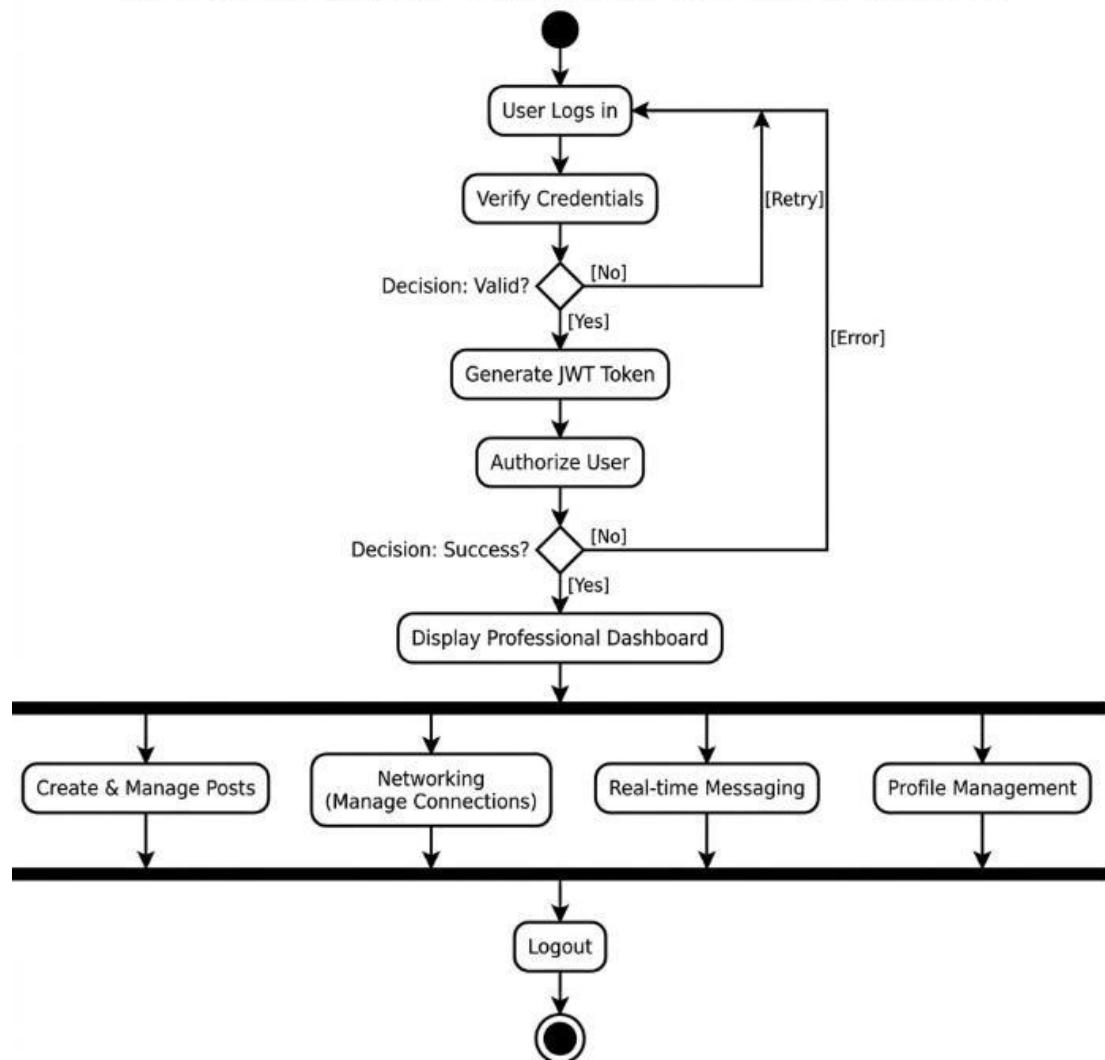
## 2. Admin Activity Diagram:

**UML Activity Diagram: Recruiter Role in Connexus**



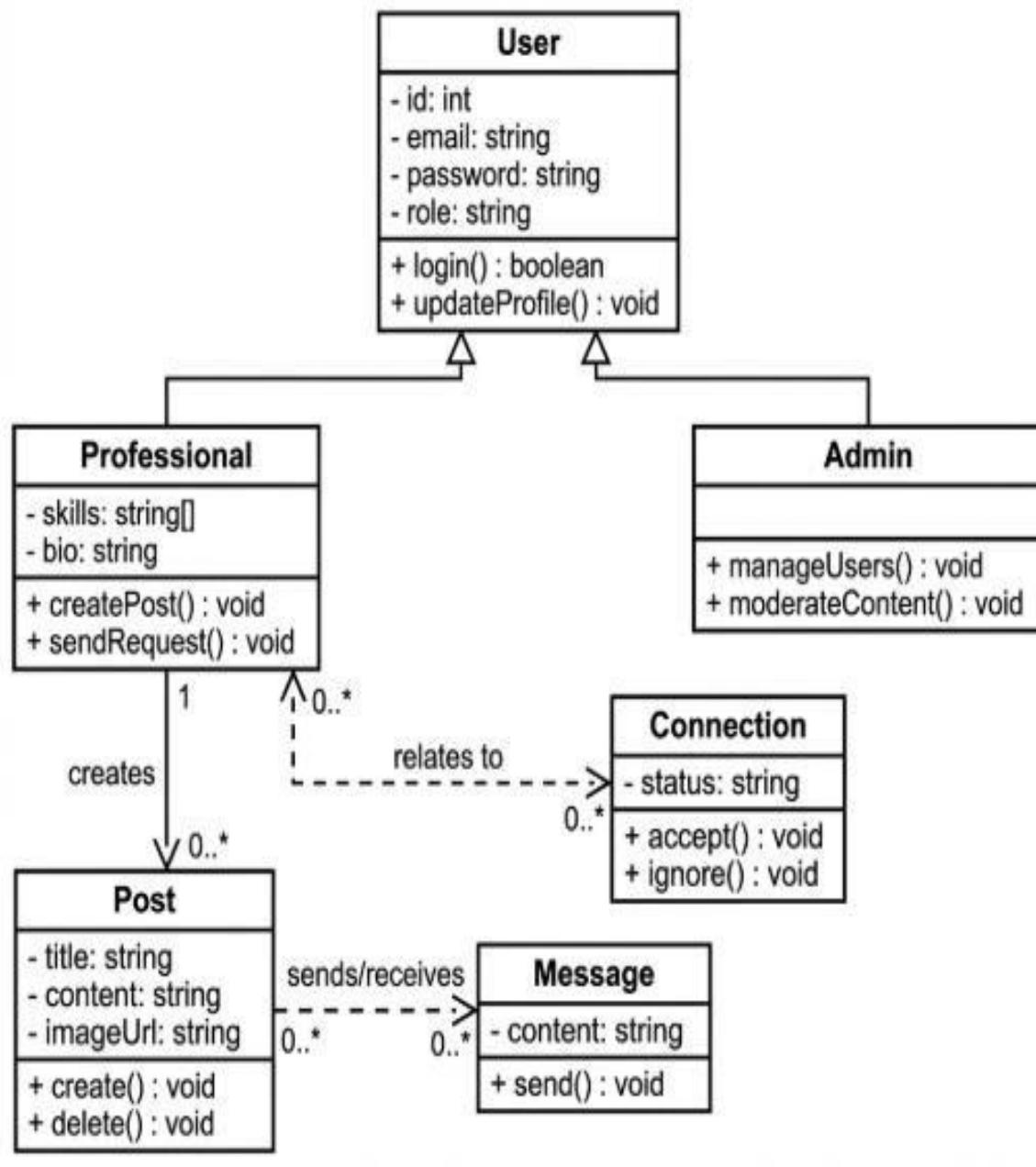
### 3. User Activity Diagram

UML Activity Diagram - Professional User Role in Connexus



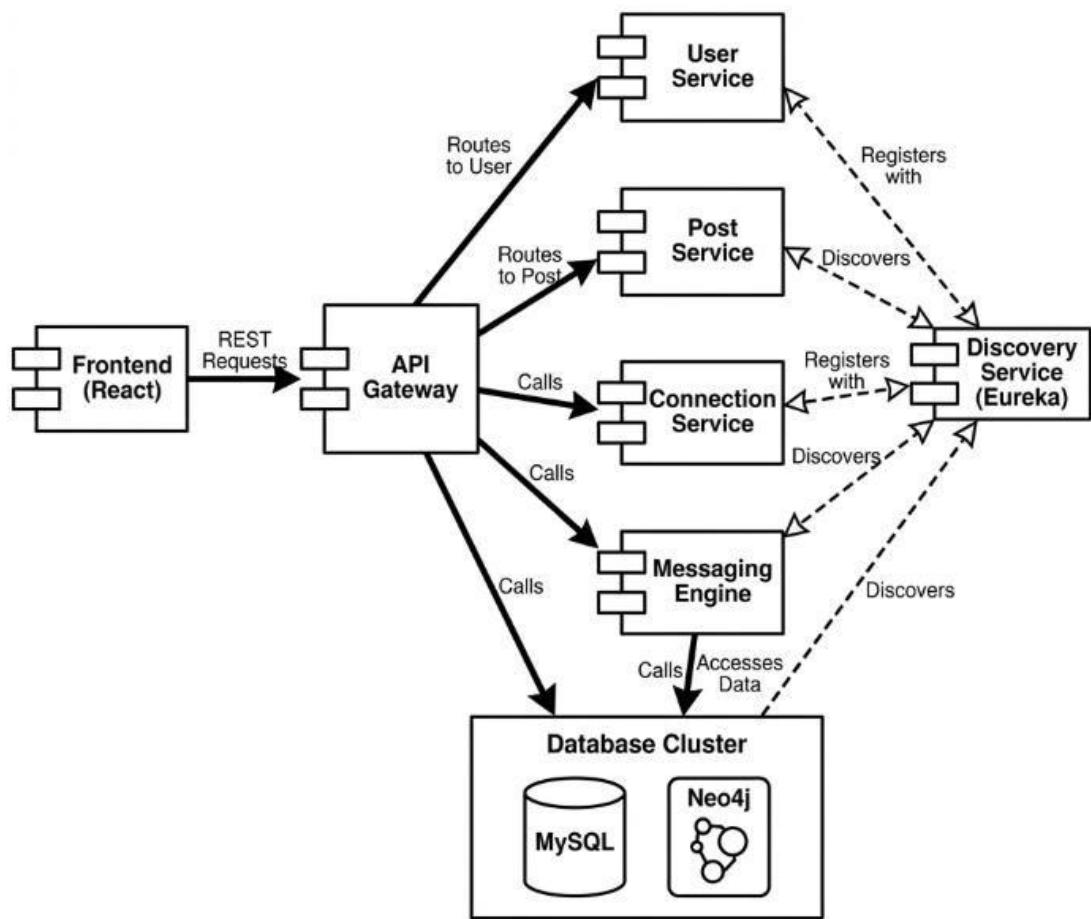
### 3.5 Class Diagram:

## UML Class Diagram: Connexus



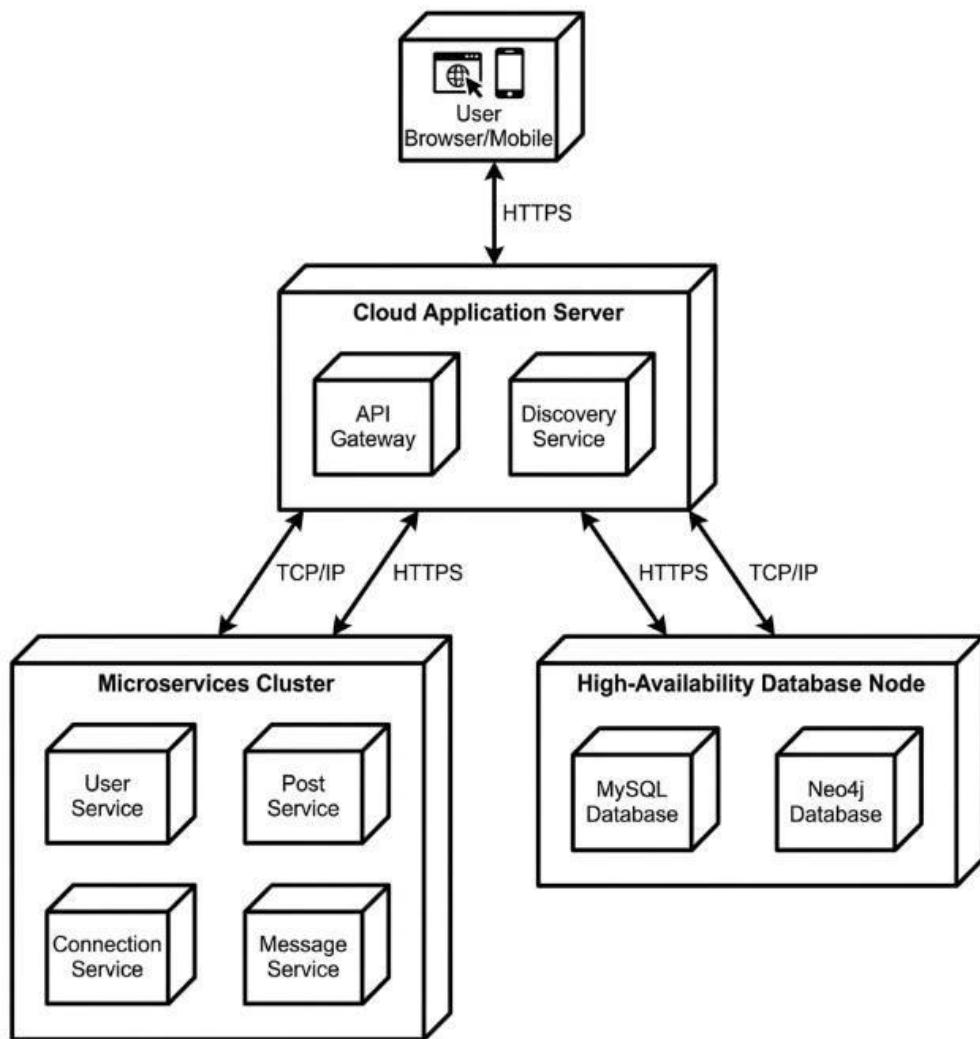
### 3.6 Component Diagram:

# Connexus



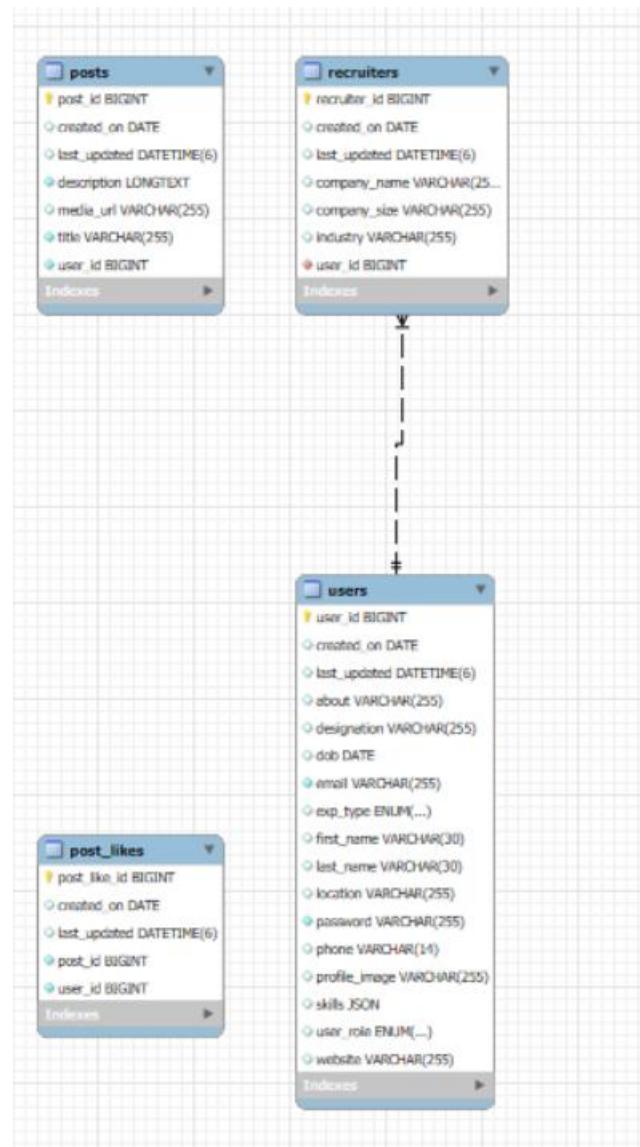
### 3.7 Sequence Diagram

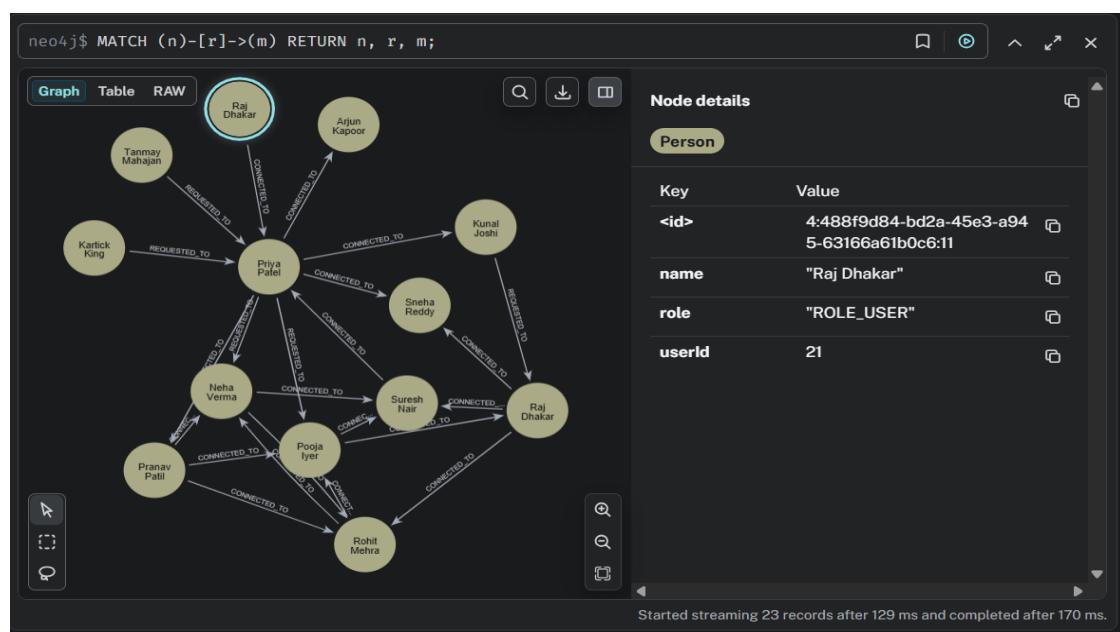
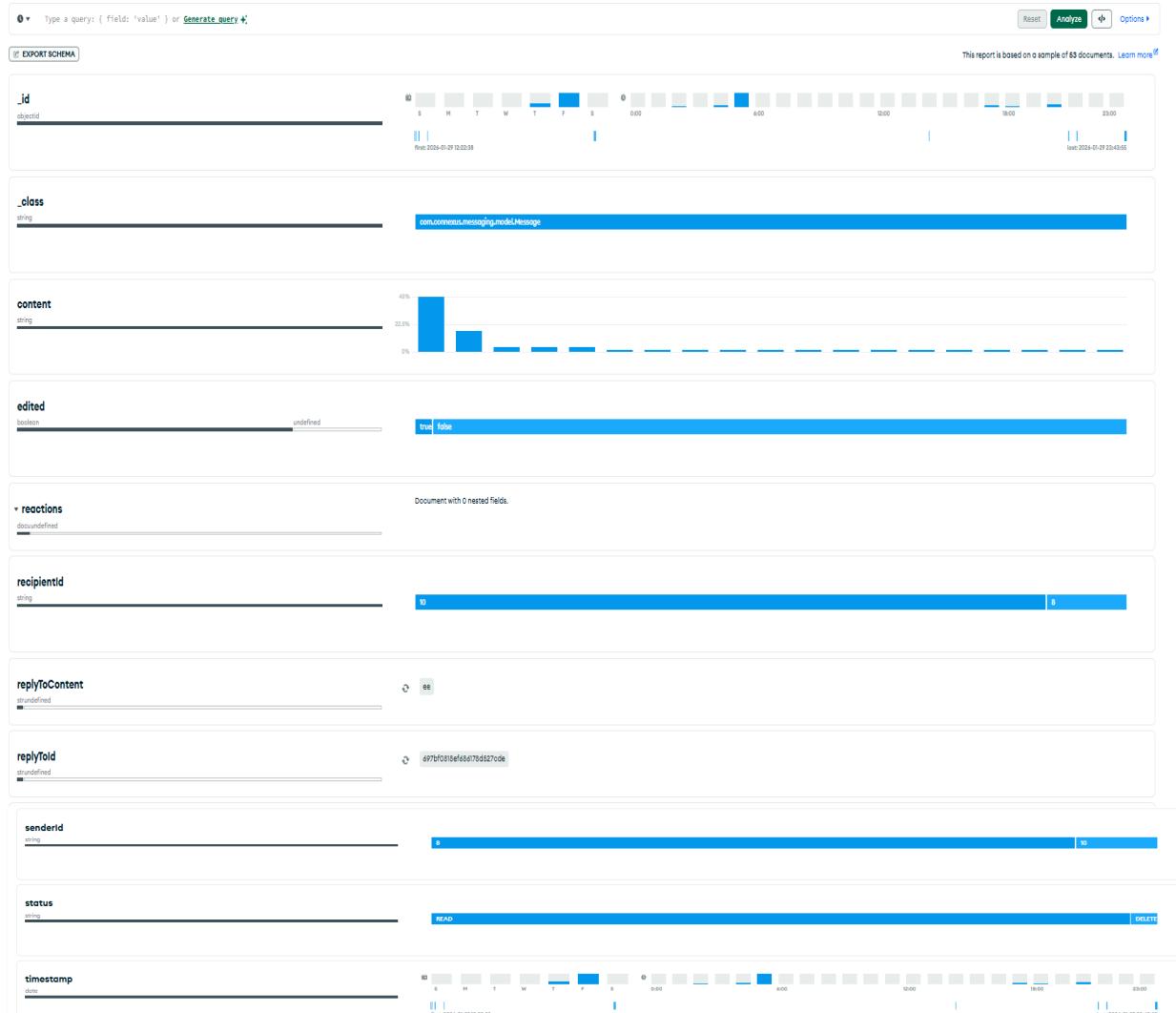
**UML Deployment Diagram: Connexus**



## 4. DATABASE DESIGN

### 4.1 Design:

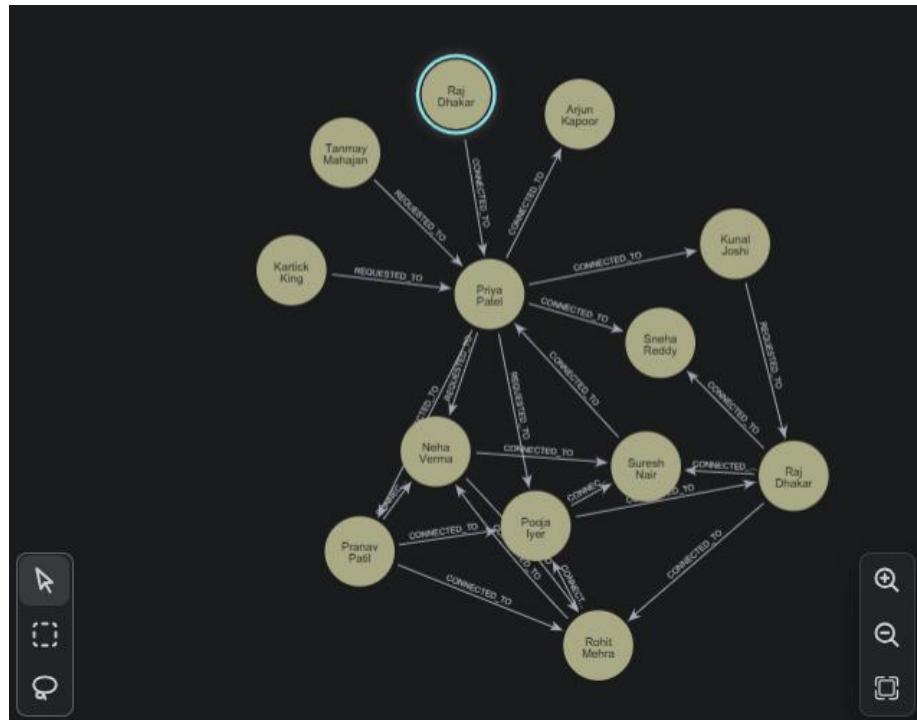




## 4.2 Tables:

The following table structures depict the database design.

## Graph Connection:



## User Table:

**Message Collection:**

```
TypeError: db.messages.findAll is not a function
connexus_chat> db.messages.find();
[
  {
    _id: ObjectId('698033b9a7bc063407b98ba8'),
    senderId: '8',
    recipientId: '10',
    content: 'hi',
    timestamp: ISODate('2026-02-02T05:18:49.052Z'),
    status: 'READ',
    edited: false,
    reactions: {},
    _class: 'com.connexus.messaging.model.Message'
  },
  {
    _id: ObjectId('698033d6a7bc063407b98ba9'),
    senderId: '8',
    recipientId: '10',
    content: 'hello',
    timestamp: ISODate('2026-02-02T05:19:18.183Z'),
    status: 'READ',
    edited: false,
    reactions: {},
    _class: 'com.connexus.messaging.model.Message'
  },
  {
    _id: ObjectId('698033dda7bc063407b98baa'),
    senderId: '8',
    recipientId: '10',
    content: 'how are you?',
    timestamp: ISODate('2026-02-02T05:19:25.439Z'),
    status: 'READ',
    edited: false,
    reactions: { '10': '😊' },
    _class: 'com.connexus.messaging.model.Message'
  },
  {
    _id: ObjectId('698035f7a7bc063407b98bab'),
    senderId: '10',
    recipientId: '8',
    content: 'I am fine.',
    timestamp: ISODate('2026-02-02T05:28:23.662Z'),
    status: 'READ',
    edited: false,
    reactions: {},
    _class: 'com.connexus.messaging.model.Message'
  },
  {
    _id: ObjectId('69803619a7bc063407b98bac'),
```

### Recruiter Table:

```
mysql> select * from recruiters;
+-----+-----+-----+-----+-----+-----+-----+
| recruiter_id | created_on | last_updated | company_name | company_size | industry | user_id |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 2024-01-12 | 2026-01-28 22:34:07.000000 | Infosys | 100,000+ | IT Services | 3 |
| 2 | 2024-01-13 | 2026-01-28 22:34:07.000000 | Tata Consultancy Services | 100,000+ | IT Consulting | 4 |
| 3 | 2024-01-14 | 2026-01-28 22:34:07.000000 | Wipro | 100,000+ | Technology | 5 |
| 4 | 2024-01-15 | 2026-01-28 22:34:07.000000 | NextGen Startup | 50-200 | Product-Based | 6 |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

### Posts Table:

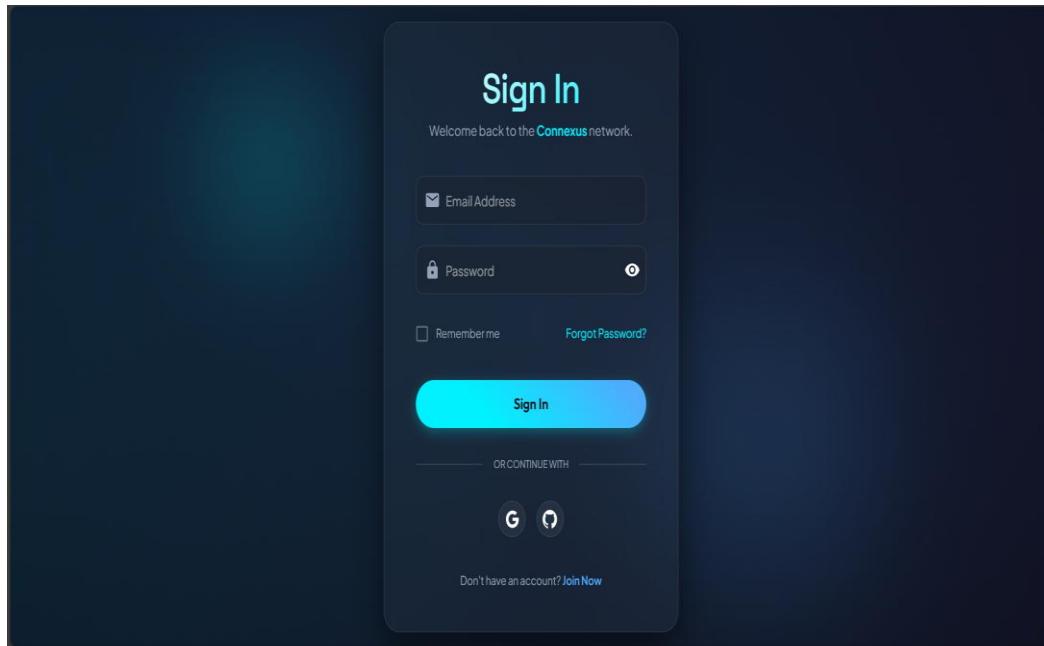
```
mysql> select * from recruiters;
+-----+-----+-----+-----+-----+-----+-----+
| recruiter_id | created_on | last_updated | company_name | company_size | industry | user_id |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 2024-01-12 | 2026-01-28 22:34:07.000000 | Infosys | 100,000+ | IT Services | 3 |
| 2 | 2024-01-13 | 2026-01-28 22:34:07.000000 | Tata Consultancy Services | 100,000+ | IT Consulting | 4 |
| 3 | 2024-01-14 | 2026-01-28 22:34:07.000000 | Wipro | 100,000+ | Technology | 5 |
| 4 | 2024-01-15 | 2026-01-28 22:34:07.000000 | NextGen Startup | 50-200 | Product-Based | 6 |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

### PostLike Table

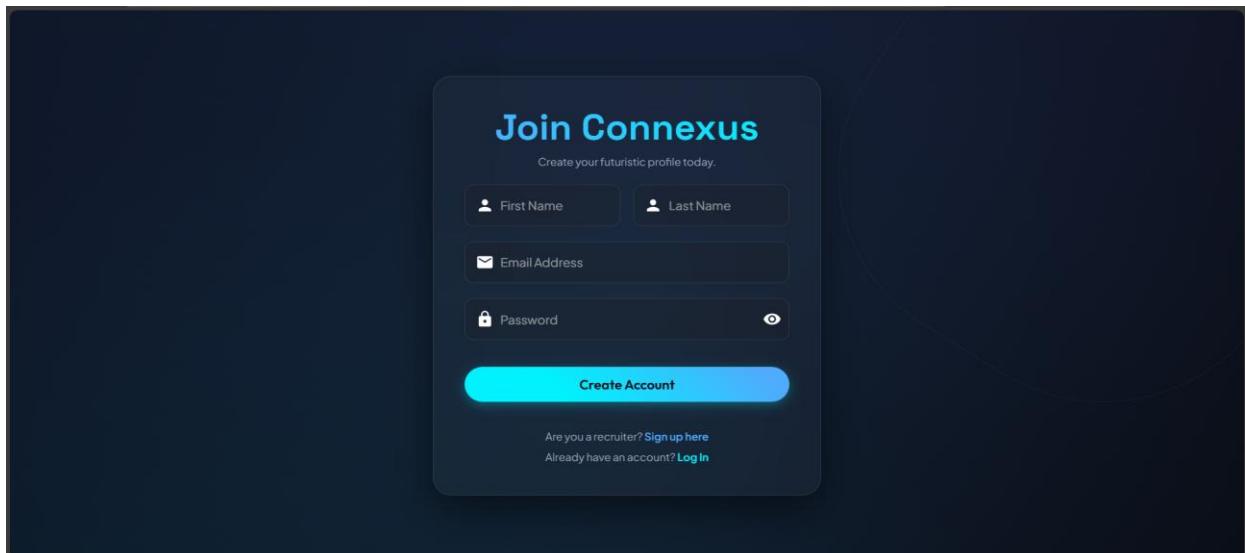
```
mysql> select * from post_likes;
+-----+-----+-----+-----+
| post_like_id | created_on | last_updated | post_id | user_id |
+-----+-----+-----+-----+
| 1 | 2024-01-05 | 2024-01-05 10:15:30.000000 | 101 | 1 |
| 2 | 2024-01-05 | 2024-01-05 10:18:12.000000 | 101 | 2 |
| 3 | 2024-01-06 | 2024-01-06 09:05:44.000000 | 102 | 1 |
| 4 | 2024-01-06 | 2024-01-06 09:10:01.000000 | 102 | 3 |
| 5 | 2024-01-07 | 2024-01-07 14:45:20.000000 | 103 | 2 |
| 6 | 2024-01-07 | 2024-01-07 15:01:55.000000 | 103 | 4 |
| 7 | 2024-01-08 | 2024-01-08 08:30:00.000000 | 104 | 1 |
| 8 | 2024-01-08 | 2024-01-08 08:31:10.000000 | 104 | 3 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

## 5. SNAPSHOTS

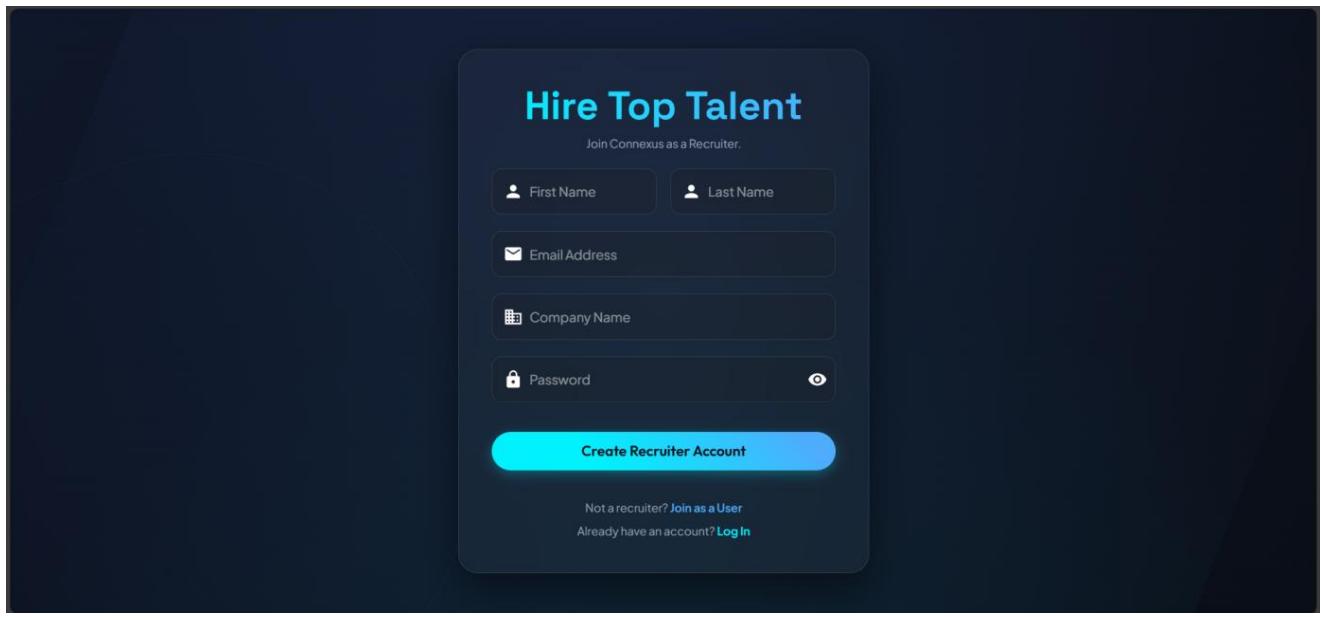
### SignIn Page:



### User SignUp Page:



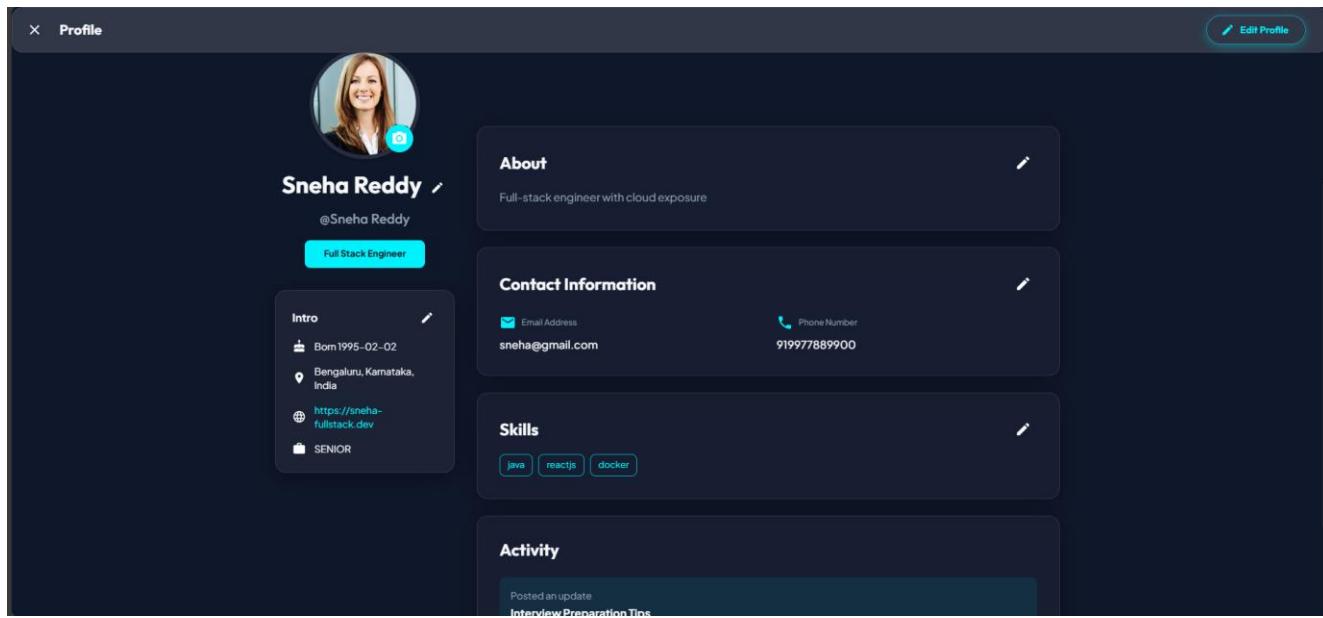
## Recruiter SignUp Page:



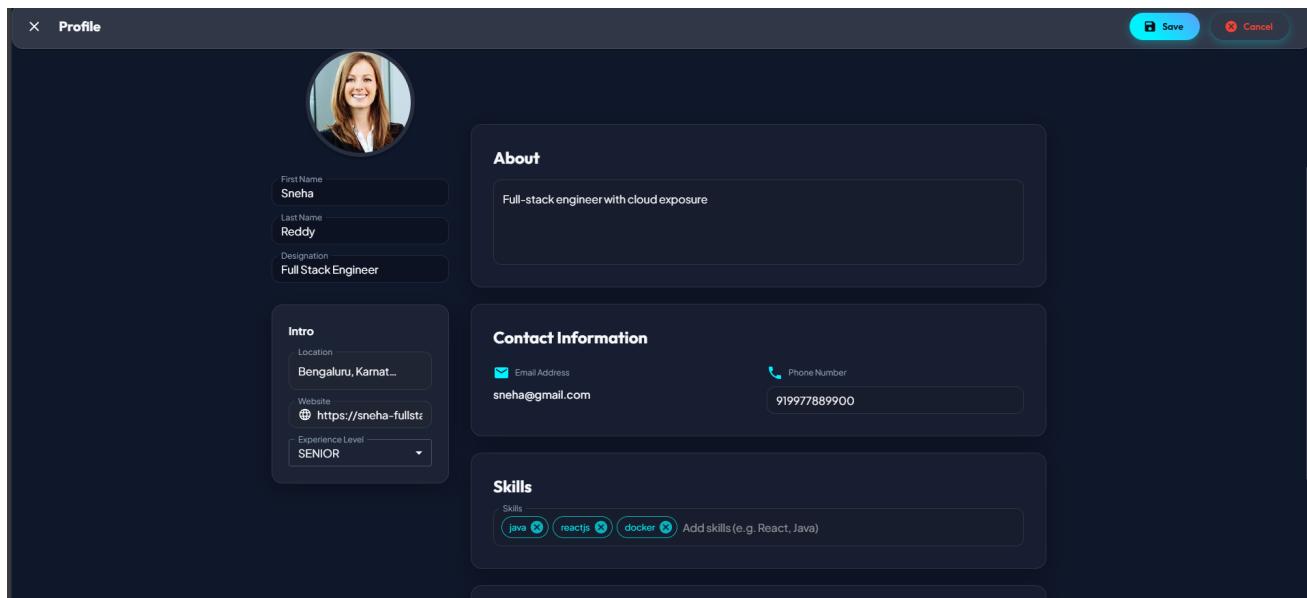
## Home Page:

The screenshot displays the Connexus home page interface. At the top, there's a navigation bar with the brand name "Connexus" and icons for Home, People, Chat, and Profile. A search bar is also present. On the left, a sidebar features a profile banner for "Sneha Reddy" (Full Stack Engineer), "Connections", and "Invitations". The main content area shows a post from "Priya Patel" (Frontend Developer) with the caption "My first post" and the text "Testing post upload from Postman". This post is overlaid on a purple night city skyline illustration with a crescent moon. To the right, there's a news feed section titled "Tech Pulse" with articles from "CNA", "YOUTUBE", "NIKKEI.COM", and "GLOBALSECURITY.ORG". The footer of the page is powered by "NewsAPI.org".

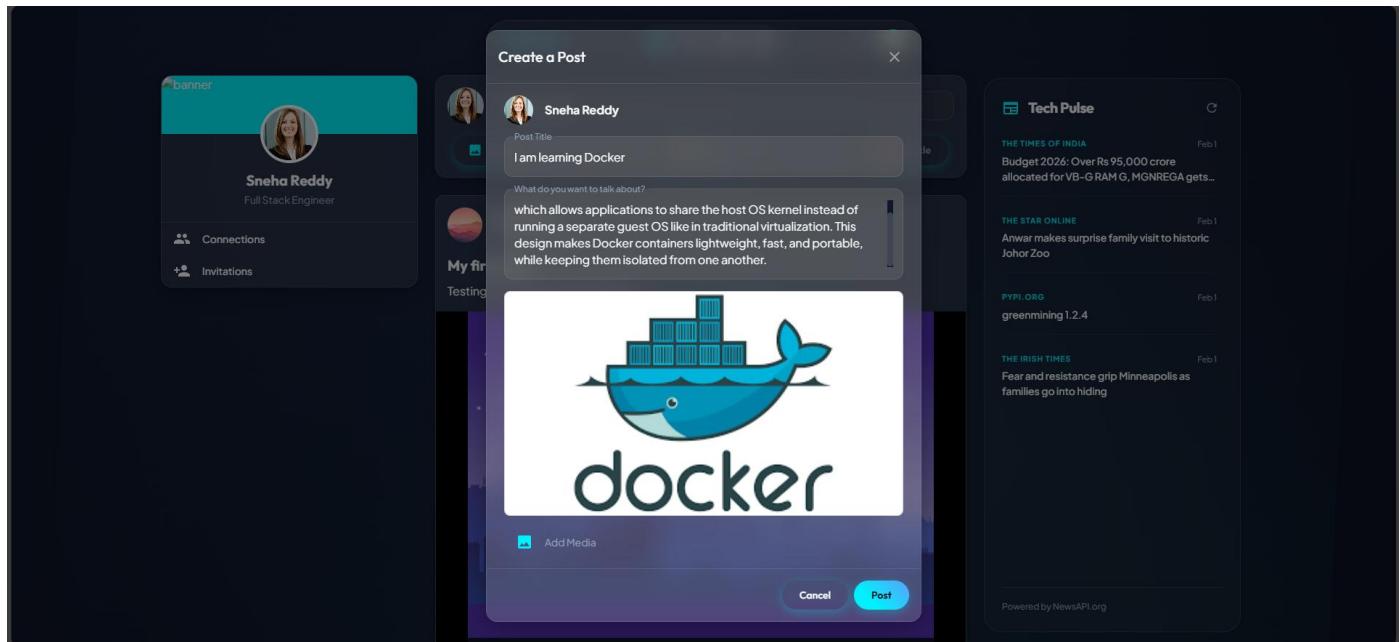
## Profile Page:



## Profile Edit Page:

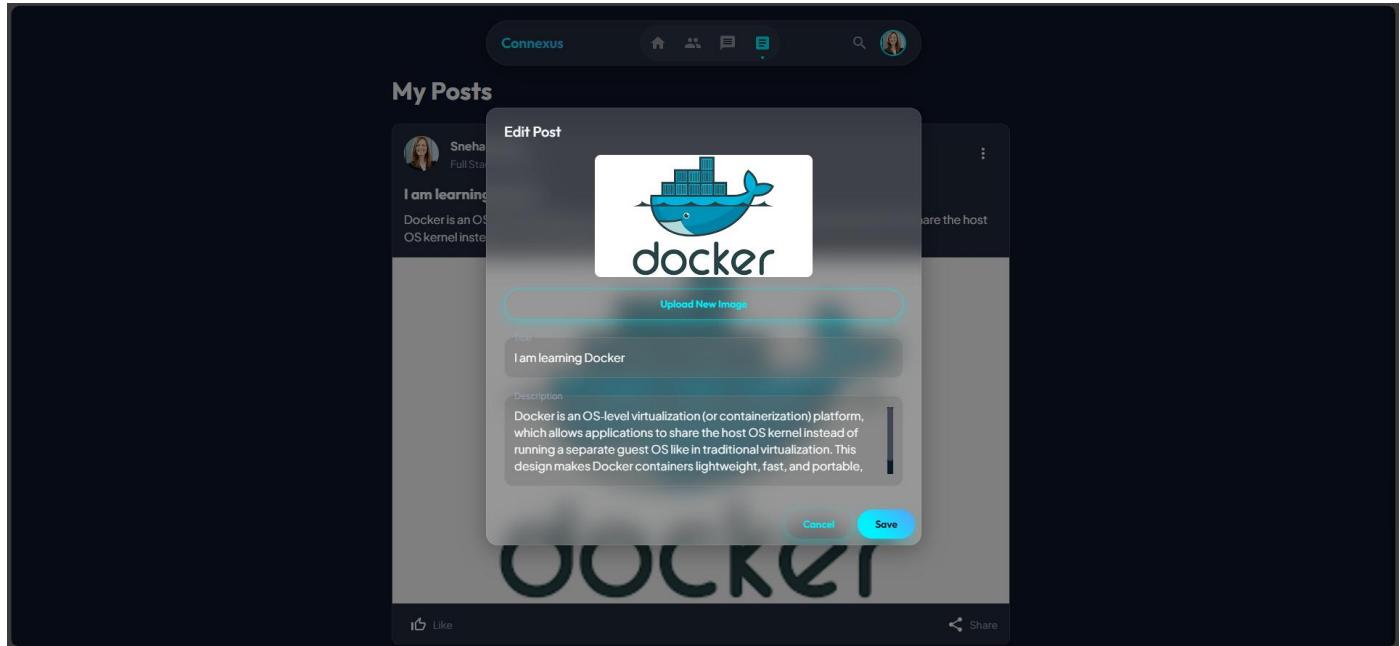


## Upload Post Image Page:



## All Post Page:



**Post Edit Page:****Invitations Page:**

### All Connections Page:

People you may know

- P Pranav Patil  
ROLE\_USER Connect
- N Neha Verma  
ROLE\_USER Connect
- R Rohit Mehra  
ROLE\_RECRUITER Connect
- P Pooja Iyer  
ROLE\_RECRUITER Connect
- S Suresh Nair  
ROLE\_RECRUITER Connect
- R Raj Dhakar  
ROLE\_USER Connect

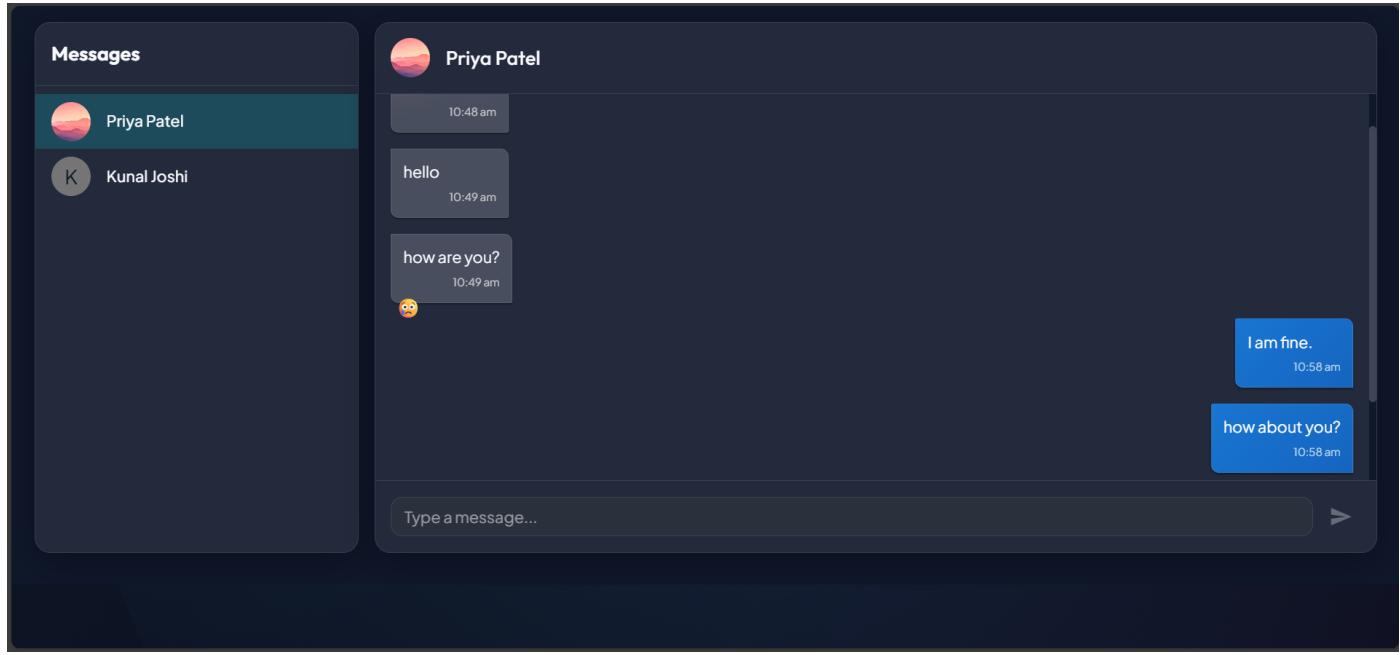
### Network Page:

Connexus

My Network

- Priya Patel  
ROLE\_USER Message
- Kunal Joshi  
ROLE\_USER Message

## Message Page:



## Recruiter\_DashBoard Page:

The screenshot shows the Connexus Recruiter Dashboard. At the top, there is a header with the Connexus logo and navigation icons. Below the header, the word "Dashboard" is prominently displayed. A welcome message reads "Welcome back, Rohit Mehra. Find your next top talent." The dashboard features four key metrics in cards:

- Total Candidates: 21
- Active Jobs: 12
- Profile Views: 1.2k
- Growth: +24%

Below these metrics is a search bar with the placeholder "Search candidates by name, email, or role...". Further down, there are three filter sections: "SKILLS" (with a dropdown menu "Select Skills"), "LOCATION" (with a dropdown menu "Select Location"), and "EXPERIENCE" (with buttons for "Junior", "Mid-Level", "Senior", "Lead", and "Manager").

## Recruiter User List:

Recent Candidates		Showing 21 candidates	
Candidate	Action		
 Pranav Patil	 Candidate		
p@gmail.com <small>spring-boot react mysql hibernate</small> Kolhapur, Maharashtra, India			
 Neha Verma	 Candidate		
neha.admin@gmail.com <small>java spring-security mysql</small> Delhi, India			
 Rohit Mehra	 Recruiter		
rohit@infosys.com <small>java spring-boot microservices docker</small> Bengaluru, Karnataka, India			
 Pooja Iyer	 Recruiter		
pooja@tcs.com <small>css html reactjs javascript</small> Chennai, Tamil Nadu, India			
 Suresh Nair	 Recruiter		
suresh@wipro.com <small>java neo4j mysql</small> Hyderabad, Telangana, India			

## Recruiter SignUp Page:

### Hire Top Talent

Join Connexus as a Recruiter.

Not a recruiter? [Join as a User](#)  
Already have an account? [Log In](#)

## Profile Edit Page:

The screenshot shows a profile edit page with a dark background. At the top right are 'Save' and 'Cancel' buttons. On the left, there's a circular profile picture of a woman with long brown hair. Below it are three input fields: 'First Name' (Sneha), 'Last Name' (Reddy), and 'Designation' (Full Stack Engineer). To the right, there are three main sections: 'About' (containing the text 'Full-stack engineer with cloud exposure'), 'Contact Information' (with 'Email Address' sneha@gmail.com and 'Phone Number' 919977889900), and 'Skills' (listing 'java', 'reactjs', and 'docker' with an 'Add skills' button).

## **6. CONCLUSION**

In conclusion, Connexus addresses key requirements for modern professional networking and recruitment systems, providing a solid foundation for further development and improvement. Its modular microservices architecture and adherence to high-security standards and real-time communication protocols position it well for continued success and adaptation in the evolving professional networking landscape.

The Connexus project exemplifies a sophisticated approach to professional social platforms by leveraging cutting-edge technologies and methodologies. This application is designed to meet the needs of both professionals and recruiters, ensuring a seamless experience for networking, content sharing, and candidate discovery while adhering to stringent security standards and performance metrics.

Its blend of modern technologies, robust security, and user-centric design positions it as a valuable tool for users seeking to enhance their networking and talent acquisition capabilities. The project's architecture and features provide a strong foundation for ongoing development, making it adaptable to future technological advancements and market trends in the professional industry.

## **7. REFERENCES**

1. <https://docs.spring.io/spring-security/reference/>
2. <https://docs.spring.io/spring-boot/index.html/>
3. Garcia-Molina, H., Ullman, J. D., & Widom, J. (2008). *Database Systems: The Complete Book*. Prentice Hall.
4. Codd, E. F. (1970). *A Relational Model of Data for Large Shared Data Banks*. Communications of the ACM, 13(6), 377-387.
5. Fowler, M. (2003). *Patterns of Enterprise Application Architecture*. Addison- Wesley.
6. Evans, E. (2003). *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley.
7. Soni, D., Nord, R. L., & Hofmeister, C. (1995). *Software Architecture in Industrial Applications*. Proceedings of the 17th International Conference on Software Engineering, 196-207.