DOCUMENTATION OF LINKED IN CLONE

NAME – KRUTI KAUSHIK MOHAPATRA

Email – krutikaushikm@gmail.com

CONTACT NO – 7205172909

- **TECH STACK – MERN STACK WEB DEVELOPMENT**
  - **MONGODB(DATA BASE)**
  - **REACT.JS(FRONTEND)**
  - **NODE.JS(BACKEND)**
  - **GITHUB(HOSTING)**

Github link - https://github.com/kruticode/Linked-In-Clone-KKM

DESCRIPTION:-

1. Help students to connect with each other

2. Login Credentials (Only Engineering students)

3. Notificaiton (Chatting) - Messaging Feature

4. Applicaiton ML (You can decide how you are going to apply ML concept)

5. Database - MongoDB

6. Announcing Internship Opportunities

(Direct access for companies , HR Team - Announce - Only HR team can announce opportunties - Not students)

LinkedIn-Clone

- **Developed user profile, connections, job listing and application modules using MERN Stack and Kafka.**
- **Implemented Redis for data caching, containerized the application services and hosted on AWS.**
- **Implemented MongoDB replication and tested the performance on JMeter and Mocha.**

Features Provided

- **User Login and Sign Up**
- **Recruiter Login and Sign Up**
- **Search People**
- **Search Jobs**
- **Connect People / Recruiters**
- **Easy Apply / Custom Apply**
- **View Connections**
- **Search Jobs based on Location, Company Name**
- **Accept / Remove / Ignore Connections**
- **Post Jobs**
- **Specific To Recruiter**
- **Send / Receive Messages**
- **View Half Filled Forms (specific to recruiter)**
- **Number of views of your post (specific to recruiter)**
- **View Clicks on your post (specific to recruiter)**
- **View Number of Applicants of your post (specific to recruiter)**
- **View Your Top Posts (specific to recruiter)**

Technologies Used

- React
- Redux
- Node
- AWS RDS MySql
- Mongo DB Cluster
- Kafka
- Plotly
- S3 Buckets
- Json Web Token

What is Kafka?

Kafka is a distributed messaging queue which is an imperative programming structure which is based event-driven programming.

Kafka Features

- Publish and subscribe messaging queue.
- Reacts to events in a real time
- High tolerance with highly available .

Kafka Semantics

- Atleast Once
- Atmost Once
- Only once (most difficult one)

Scaling Techniques

- Mongo DB Cluster with Replica sets
- Auto Scaling groups with Elastic Load Balancing
- AWS RDS Cluster

Installation Requirements

For development, you will only need Node.js installed on your environement. And please use the appropriate Editorconfig plugin for your Editor (not mandatory).

Node

Node is really easy to install & now include NPM. You should be able to run the following command after the installation procedure below.

$ node --version

v0.10.24

$ npm --version

1.3.21

Node installation on Linux

sudo apt-get install python-software-properties

sudo add-apt-repository ppa:chris-lea/node.js

sudo apt-get update

sudo apt-get install nodejs

Node installation on Windows

Just go on official Node.js website & grab the installer. Also, be sure to have git available in your PATH, npm might need it.

Front End

cd LinkedIn-Clone/linkedin-frontend

npm install

BACKEND

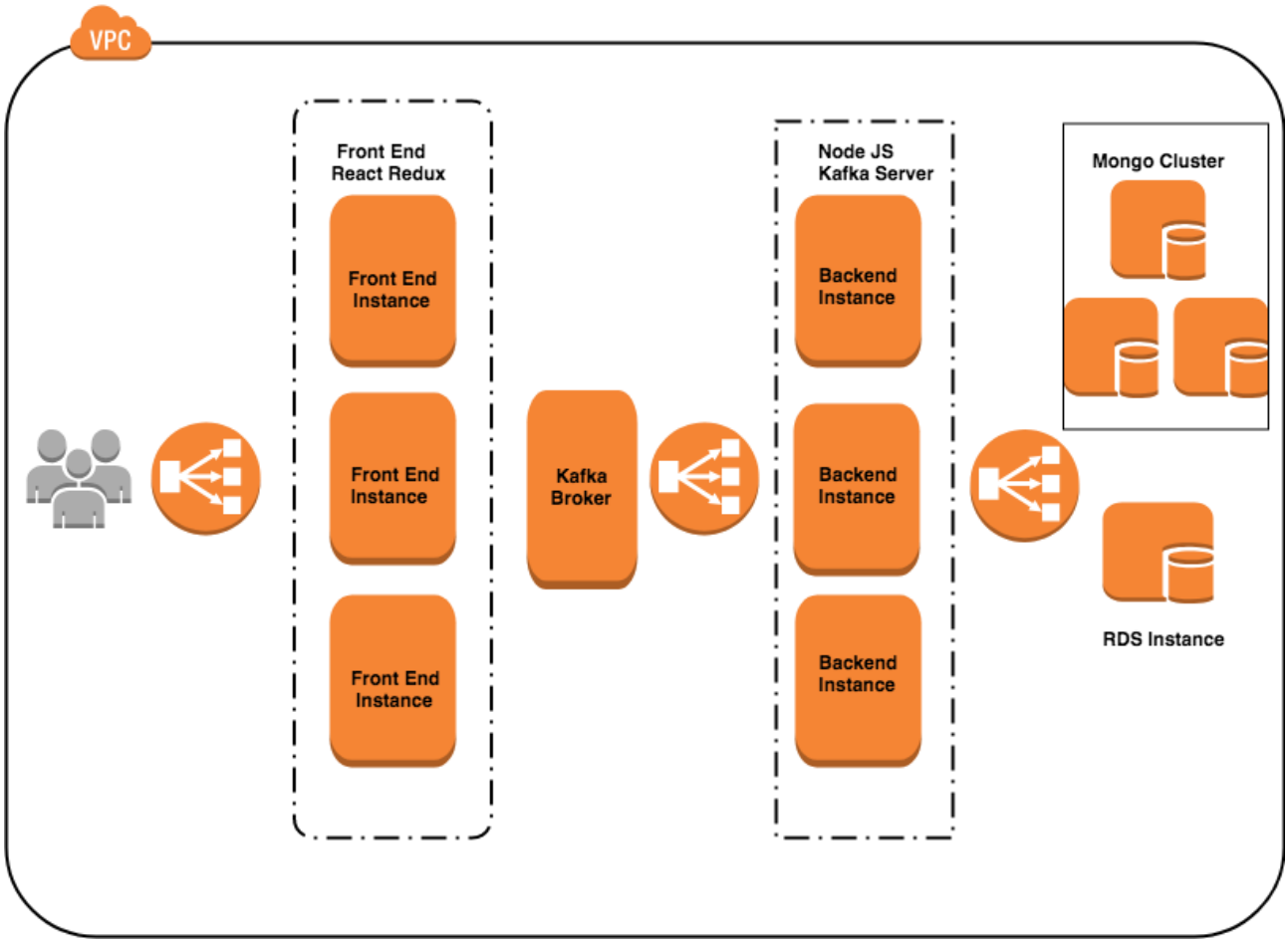
cd LinkedIn-Clone/linkedin-backend/

npm install

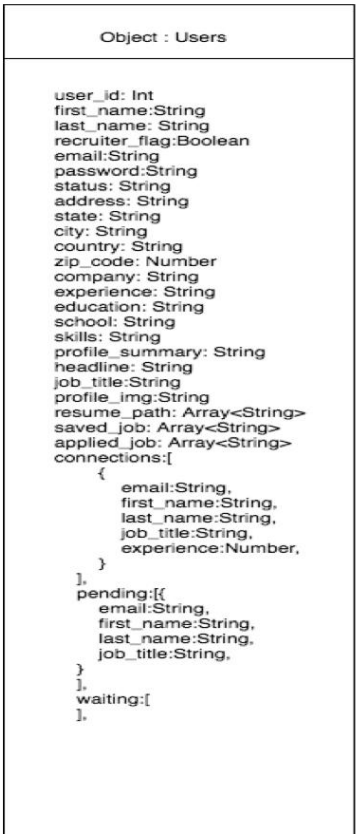
Start & watch

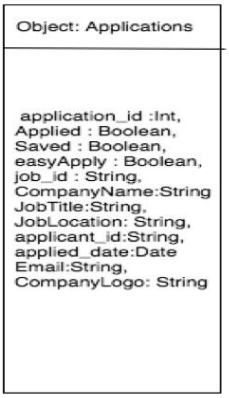
npm start

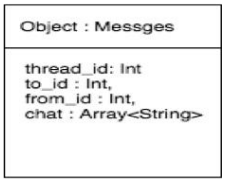
Application Architecture Diagram

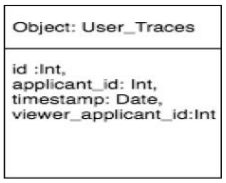


Database Schema

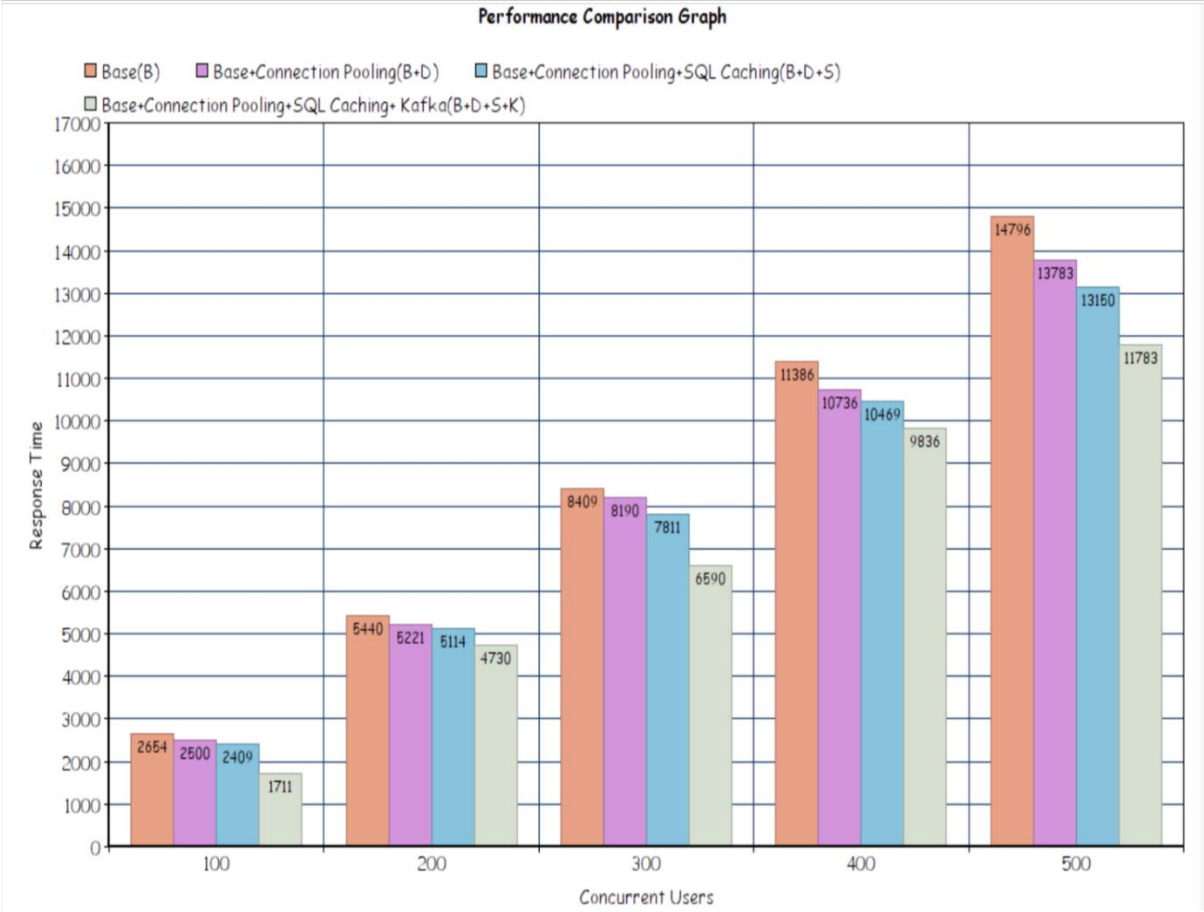


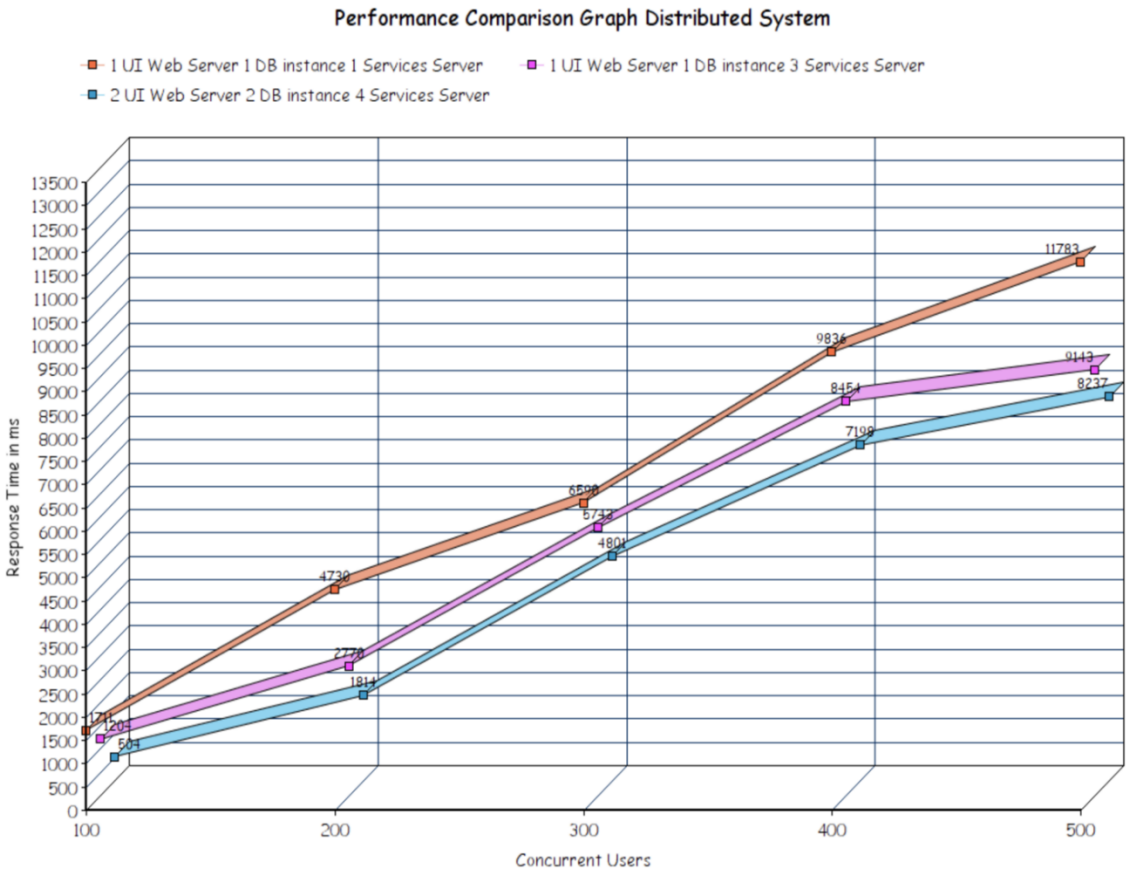






Performace Trace (JMeter Testing)




Performance Comparison Graph Distributed System
1 UI Web Server 1 DB instance 1 Services Server
1 UI Web Server 1 DB instance 3 Services Server
2 UI Web Server 2 DB instance 4 Services Server
Response Time in ms
Concurrent Users
100
200
300
400
500
11783
9836
9143
8454
8237
7198
6598
5732
4801
4730
2770
1811
1721
1205
504