Design Document for The Hive App

Group 4-Kabir-8

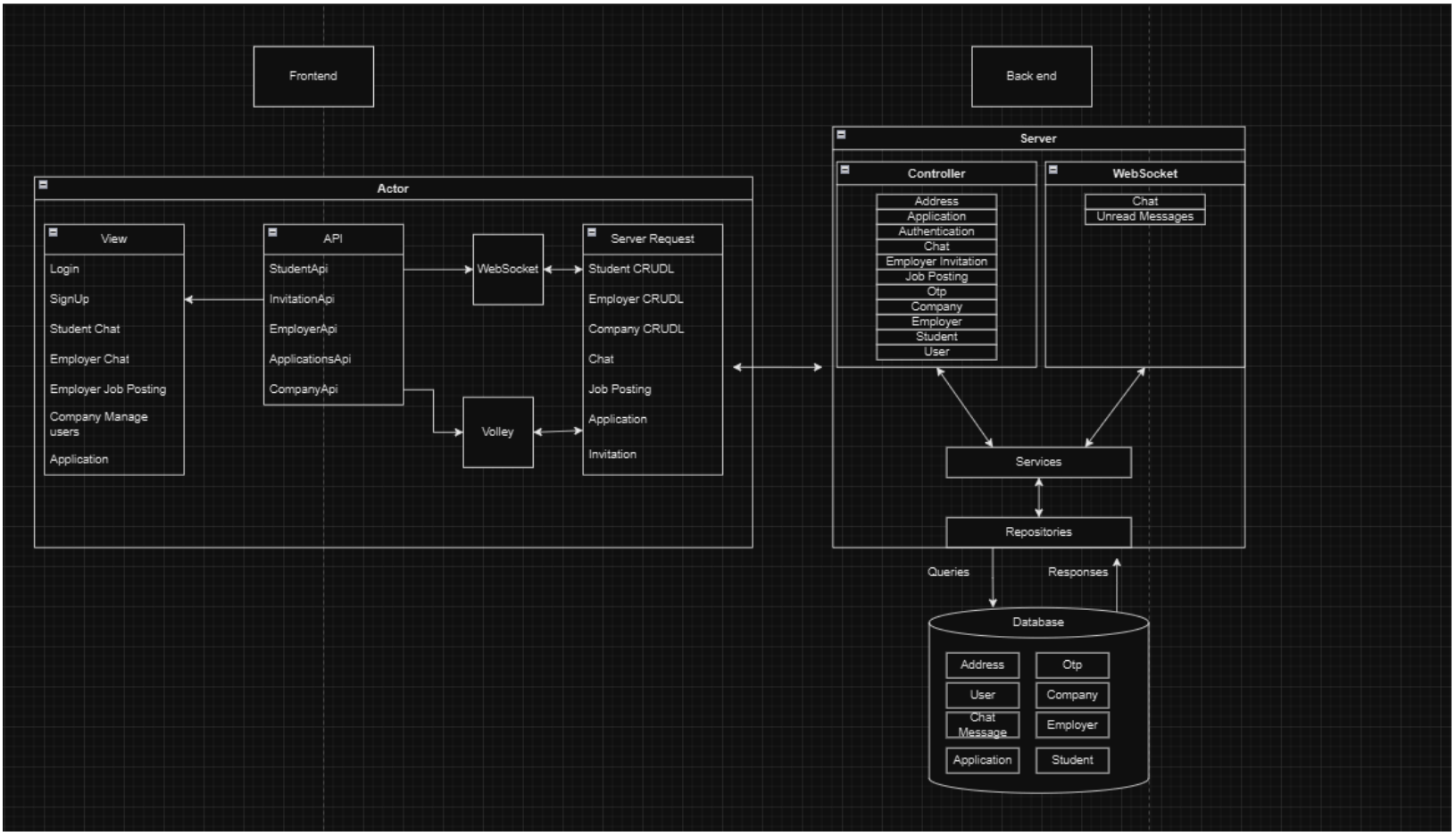
Member1 Name: Adeife 25% contribution

Member2 Name: Alisala 25% contribution

Member3 Name: Lucas Sorge 25% contribution

Member4 Name: Lucas Rillo 25% contribution

Block Diagram



Design Description

Login: The user is able to login with credentials made in the signup, when the user has forgotten their password, the user is able to create a new password by using the “forgot password” prompt.

SignUp: The user can sign up to the Hiveap based on the chosen user identification being student, Employer or the company.

User: the user is identified by either the student, employer or the company, these are user choices when signing up for HIveapp.

Student chat: After matching with an employer, students can communicate directly to discuss job opportunities.A real time communication once the student is accepted for the job applied by the employer.

Employer chat: Functionality to initiate and maintain conversations with student users after a job match is made.

Employer Job Posting: This allows employers to create new job listings that students can view and apply for.Employers are able to post new jobs, update job details, remove jobs and retrieve a list of jobs.

Company Manage User: This allows the company to add new employers, update employer details, remove employers, and retrieve a list of employers.

Employer Invitation: The company has the implemented functionality to create, modify, remove, and view invites sent to employers.The company keeps track of who has been invited for employer position.

Application: The student has the ability to swipe through posted jobs by either applying or skipping. whereas the employer either accepts or rejects the student based on qualified details after viewing their details.

Address: Each user has an address implemented within the user’s information, which consists of enum for user choices, null values, numbers and words for specified address information.

Authentication: The authorization allows a safer passage for information to be retrieved from the backend by requesting to the backend with the specified authorized details.

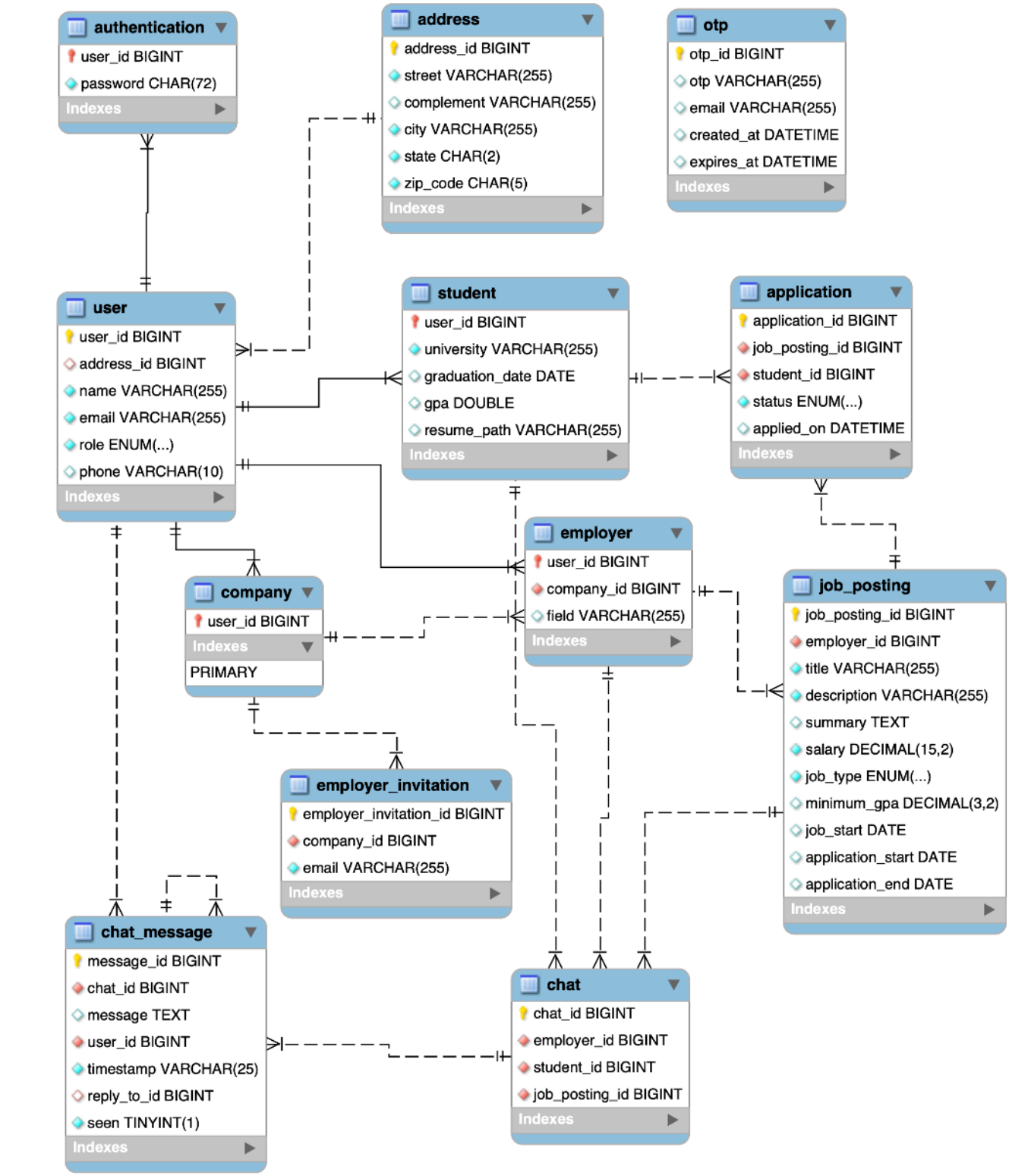
Otp: Whenever the user forgets their password, they are able to send a one time request to the backend to retrieve their new password by creating a new password.

Chat: for any user, they are able to communicate in real time with the intended user, allowing any user to effectively get their message across.

Unread messages: A websocket that enables the user to see real time unread messages within each chat. Unread messages makes it easier for either user to see any unread messages from whom that they are talking to.

User API: All the apis allow for easier transition to get, update,update, post information from the backend.

Tables


authentication
user_id BIGINT
password CHAR(72)
Indexes
address
address_id BIGINT
street VARCHAR(255)
complement VARCHAR(255)
city VARCHAR(255)
state CHAR(2)
zip_code CHAR(5)
Indexes
otp
otp_id BIGINT
otp VARCHAR(255)
email VARCHAR(255)
created_at DATETIME
expires_at DATETIME
Indexes
user
user_id BIGINT
address_id BIGINT
name VARCHAR(255)
email VARCHAR(255)
role ENUM(...)
phone VARCHAR(10)
Indexes
student
user_id BIGINT
university VARCHAR(255)
graduation_date DATE
gpa DOUBLE
resume_path VARCHAR(255)
Indexes
application
application_id BIGINT
job_posting_id BIGINT
student_id BIGINT
status ENUM(...)
applied_on DATETIME
Indexes
employer
user_id BIGINT
company_id BIGINT
field VARCHAR(255)
Indexes
company
user_id BIGINT
Indexes
PRIMARY
job_posting
job_posting_id BIGINT
employer_id BIGINT
title VARCHAR(255)
description VARCHAR(255)
summary TEXT
salary DECIMAL(15,2)
job_type ENUM(...)
minimum_gpa DECIMAL(3,2)
job_start DATE
application_start DATE
application_end DATE
Indexes
employer_invitation
employer_invitation_id BIGINT
company_id BIGINT
email VARCHAR(255)
Indexes
chat_message
message_id BIGINT
chat_id BIGINT
message TEXT
user_id BIGINT
timestamp VARCHAR(25)
reply_to_id BIGINT
seen TINYINT(1)
Indexes
chat
chat_id BIGINT
employer_id BIGINT
student_id BIGINT
job_posting_id BIGINT
Indexes