

# ScreenSketches

4\_Kabir\_8

***The Hive App***

Lucas Martins Sorge

Adeife Fadahunsi

Lucas Nerone Rillo

Alisala Mwamba

# Actors and Corresponding Functionalities

## 1. Company

- **Manage Employers (Add/Update/Delete/Get):** Allows the company to add new employers, update employer details, remove employers, and retrieve a list of employers.
- **Manage Invitations (Add/Update/Delete/Get):** Functionality to create, modify, remove, and view invites sent to employers.

## 2. Admin

- **Manage Users (Add/Update/Delete/Get):** Admin has the ability to add, update, delete, and view all users (students, employers, and companies).

## 3. Employer

- **Add Job Posts:** Allows employers to create new job listings that students can view and apply for.
- **Chat with Students:** Functionality to initiate and maintain conversations with student users after a job match is made.
- **Manage Applications:** Employers can view and track applications from students, allowing them to filter, review, and select candidates.

## 4. Student User

- **Swipe for Jobs (Apply/Pass):** Students can swipe right to apply for jobs they are interested in or swipe left to pass on a job posting, similar to the Tinder app interface.
- **Chat with Employers:** After matching with an employer, students can communicate directly to discuss job opportunities.
- **Track Applications:** Students can view the status of their applications, including jobs they've applied for and where they stand in the process.

# Non-Functional Requirements

1. **Performance (Response Time):** The system should provide quick response times, ensuring that job swiping, chats, and other interactions happen in real time without noticeable delays.
2. **Scalability:** The platform must be able to handle an increasing number of users (students, employers, admins) and growing data (job listings, chat history, applications) without degradation in performance.
3. **Reliability:** The system should be highly reliable, with minimal downtime, ensuring users can access job postings, apply for jobs, and communicate with each other at any time without interruptions.

# Tables and Fields

```
-- TABLES
-----
CREATE TABLE address (
  address_id BIGINT PRIMARY KEY AUTO_INCREMENT,
  street VARCHAR(255) NOT NULL,
  complement VARCHAR(255) NULL,
  city VARCHAR(255) NOT NULL,
  state CHAR(2) NOT NULL,
  zip_code CHAR(5) NOT NULL
);
CREATE TABLE user (
  user_id BIGINT PRIMARY KEY AUTO_INCREMENT,
  address_id BIGINT UNIQUE NULL,
  name VARCHAR(255) NOT NULL,
  email VARCHAR(255) UNIQUE NOT NULL,
  role ENUM ('student', 'employer', 'admin', 'company') NOT NULL,
  phone VARCHAR(10) NULL,
  FOREIGN KEY (address_id) REFERENCES address(address_id) ON DELETE SET NULL
);
CREATE TABLE authentication (
  user_id BIGINT PRIMARY KEY,
  password CHAR(72) NOT NULL,
  FOREIGN KEY (user_id) REFERENCES user(user_id) ON DELETE CASCADE
);
CREATE TABLE company (
  user_id BIGINT PRIMARY KEY,
  FOREIGN KEY (user_id) REFERENCES user(user_id) ON DELETE CASCADE
);
CREATE TABLE employer (
  user_id BIGINT PRIMARY KEY,
  company_id BIGINT NOT NULL,
  birth_date DATE NULL,
  FOREIGN KEY (user_id) REFERENCES user(user_id) ON DELETE CASCADE,
  FOREIGN KEY (company_id) REFERENCES company(user_id) ON DELETE CASCADE
);
CREATE TABLE student (
  user_id BIGINT PRIMARY KEY,
  university VARCHAR(255) NOT NULL,
  graduation_date DATE NULL,
  birth_date DATE NULL,
  resume_path VARCHAR(255) NULL,
  FOREIGN KEY (user_id) REFERENCES user(user_id) ON DELETE CASCADE
);
CREATE TABLE admin (
  user_id BIGINT PRIMARY KEY,
  FOREIGN KEY (user_id) REFERENCES user(user_id) ON DELETE CASCADE
);
CREATE TABLE employer_invitation (
  employer_invitation_id BIGINT PRIMARY KEY AUTO_INCREMENT,
  company_id BIGINT NOT NULL,
  email VARCHAR(255) NOT NULL,
  FOREIGN KEY (company_id) REFERENCES company(user_id) ON DELETE CASCADE
);
CREATE TABLE job_posting (
  job_posting_id BIGINT PRIMARY KEY AUTO_INCREMENT,
  employer_id BIGINT NOT NULL,
  title VARCHAR(255) NOT NULL,
  description TEXT,
  summary VARCHAR(511) NOT NULL,
  salary DECIMAL(15,2) NOT NULL,
  job_type ENUM ('POST_GRADUATION_JOB', 'CO_OP', 'INTERNSHIP') NOT NULL,
  minimum_gpa DECIMAL(3, 2) NULL,
  job_start DATE NULL,
  application_start DATE NULL,
  application_end DATE NULL,
  FOREIGN KEY (employer_id) REFERENCES employer(user_id) ON DELETE CASCADE
```

```

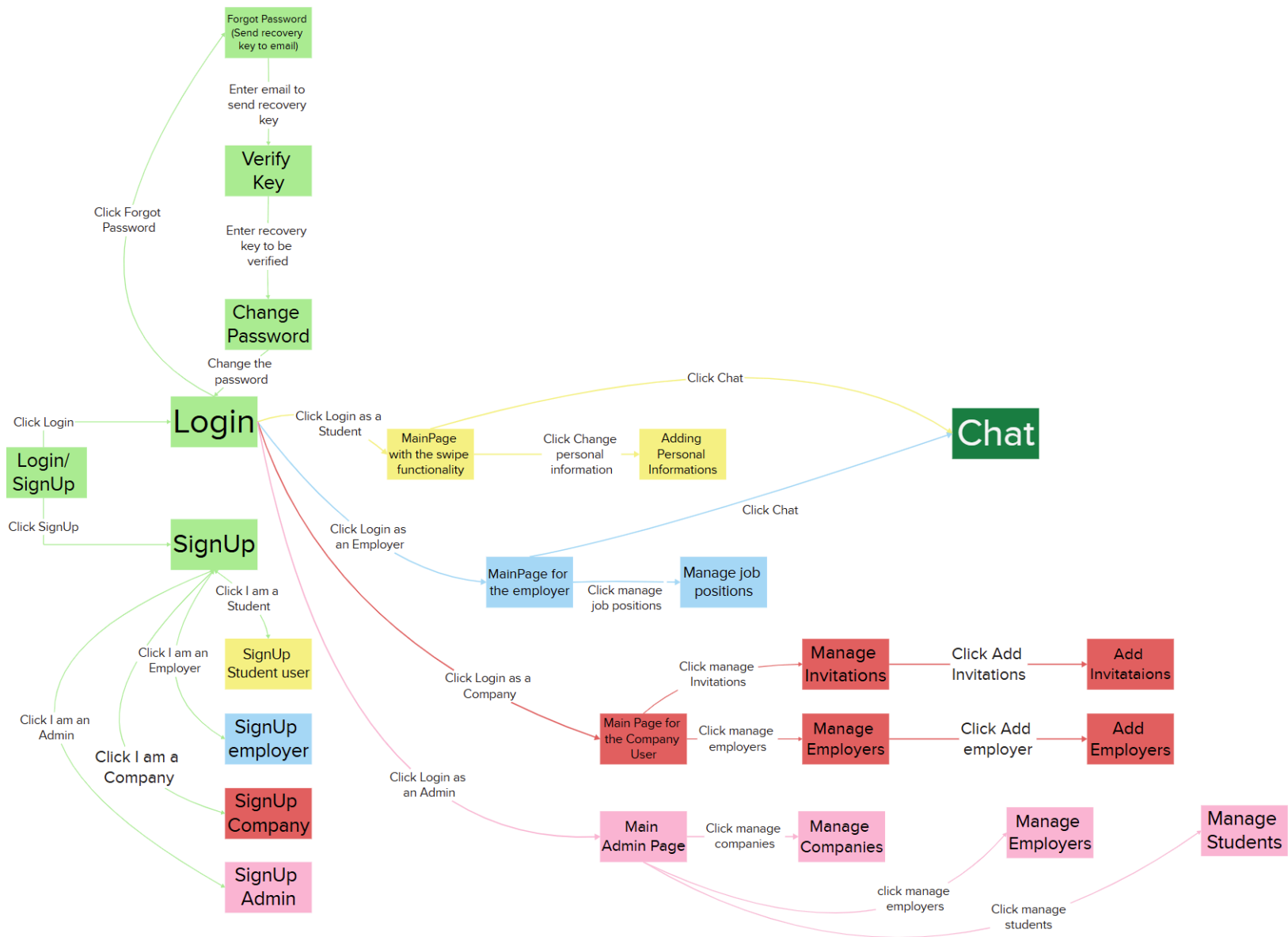
);
CREATE TABLE chat (
    chat_id BIGINT PRIMARY KEY AUTO_INCREMENT,
    employer_id BIGINT NOT NULL,
    student_id BIGINT NOT NULL,
    FOREIGN KEY (employer_id) REFERENCES employer(user_id) ON DELETE CASCADE,
    FOREIGN KEY (student_id) REFERENCES student(user_id) ON DELETE CASCADE
);
CREATE TABLE chat_message (
    message_id BIGINT PRIMARY KEY AUTO_INCREMENT,
    chat_id BIGINT NOT NULL,
    message TEXT,
    user_id BIGINT NOT NULL,
    FOREIGN KEY (chat_id) REFERENCES chat(chat_id) ON DELETE CASCADE,
    FOREIGN KEY (user_id) REFERENCES user(user_id) ON DELETE CASCADE
);
-- TRIGGERS
-----

-- Update the role based on the table inserting the data
DELIMITER $$
CREATE PROCEDURE set_user_role(IN p_user_id BIGINT, IN p_role VARCHAR(255))
BEGIN
    UPDATE user
    SET role = p_role
    WHERE user_id = p_user_id;
END$$
DELIMITER ;
-- Call the procedure to set the role to 'company'
DELIMITER $$
CREATE TRIGGER set_company_role
BEFORE INSERT ON company
FOR EACH ROW
BEGIN
    CALL set_user_role(NEW.user_id, 'company');
END$$
DELIMITER ;
-- Call the procedure to set the role to 'employer'
DELIMITER $$
CREATE TRIGGER set_employer_role
BEFORE INSERT ON employer
FOR EACH ROW
BEGIN
    CALL set_user_role(NEW.user_id, 'employer');
END$$
DELIMITER ;
-- Call the procedure to set the role to 'student'
DELIMITER $$
CREATE TRIGGER set_student_role
BEFORE INSERT ON student
FOR EACH ROW
BEGIN
    CALL set_user_role(NEW.user_id, 'student');
END$$
DELIMITER ;
-- Call the procedure to set the role to 'admin'
DELIMITER $$
CREATE TRIGGER set_admin_role
BEFORE INSERT ON admin
FOR EACH ROW
BEGIN
    CALL set_user_role(NEW.user_id, 'admin');
END$$
DELIMITER ;

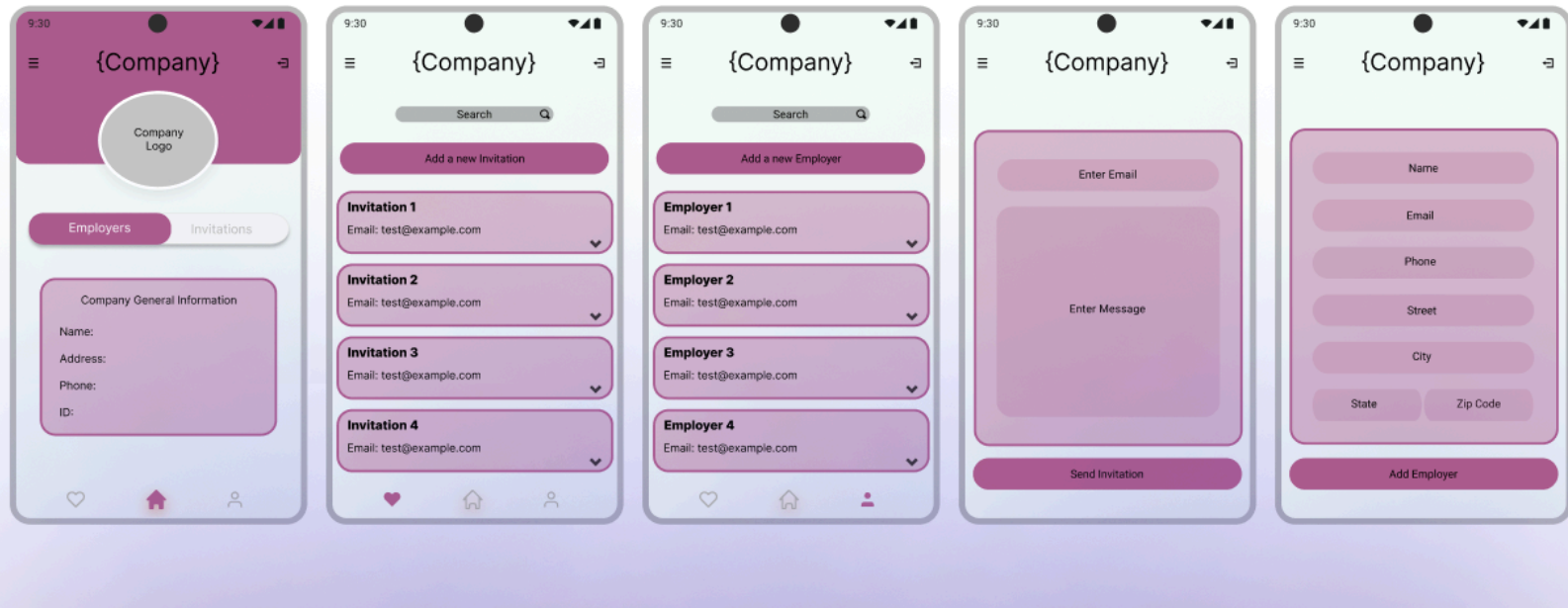
```

- We will also store students' resumes and user's profile photos, but haven't decided yet the best way to store these files.

# Screen Flow Diagram



# User “Company” Pages - Lucas Martins Sorge



## Screen 1: Dashboard / Home Screen

- This is the main dashboard for the "Company" user. It displays a company logo and two options for navigation: "Employers" and "Invitations."
- It shows general information about the company, including the name, address, phone, and ID.
- The bottom navigation bar provides quick access to other sections of the app.

## Screen 2: Invitation Management

- This screen allows the "Company" user to manage invitations.
- The search bar at the top lets the user search for specific invitations.
- Below the search bar, the user can click on the "Add a new Invitation" button to create a new invitation.
- The list shows multiple invitations with an email for each, allowing the user to view and manage existing invitations.

## Screen 3: Employer Management

- Similar to the Invitation Management screen, this one allows the "Company" user to manage employers.
- The search bar at the top lets the user search for specific employers.

- Below the search bar, the user can click on the "Add a new Employer" button to add a new employer.
- The list displays several employers, each showing their email. The user can view and manage existing employers.

#### **Screen 4: Create Invitation**

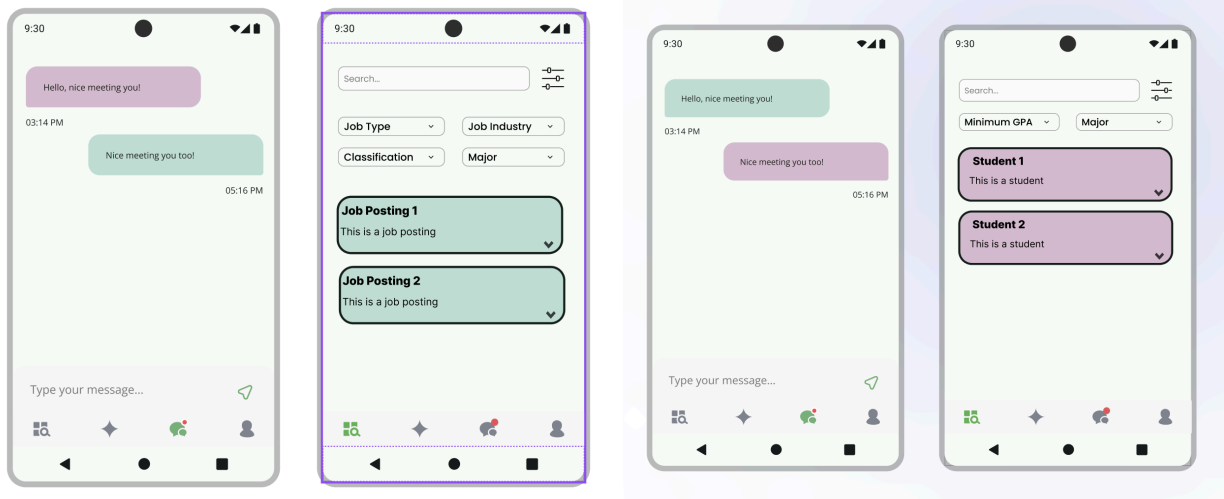
- This screen is used to send a new invitation.
- The "Company" user inputs an email address and a message.
- After filling out the details, the user can click the "Send Invitation" button, which sends the invitation to the prospective employer.

#### **Screen 5: Add Employer**

- This screen allows the "Company" user to add a new employer to their system.
- The user enters details such as name, email, phone, street, city, state, and zip code.
- Once all information is provided, the user can click the "Add Employer" button to register the new employer in the system.



# Chat and Advanced Search Pages - Lucas Nerone Rillo



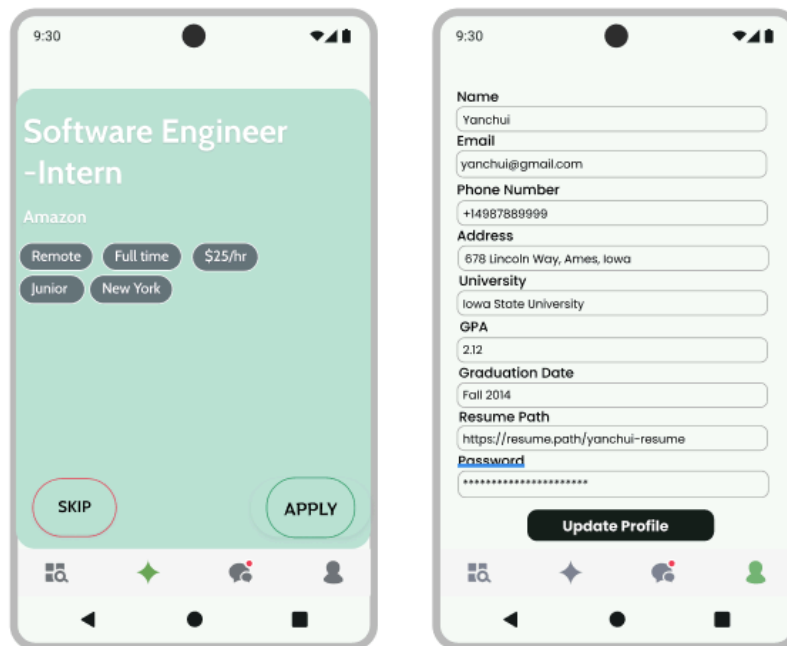
## Screens 1 and 3: Chat

- A chat UI for students to talk to employers.
- Screen 1 shows the student's perspective, which is very similar to screen 3 (employer's perspective), but colors change to match the user role.
- The bottom text bar allows the user to send messages.
- Users can click on the messages to delete them.

## Screens 2 and 4: Advanced Search

- A way for students to search employers and vice-versa on the database.
- Screen 2 shows the student's perspective, while screen 4 shows the employer's. Filters are different for each of the two (job type, job industry, ... vs student's GPA, student's major), as well as the page's color palette.
- The search bar on top works as a basic search, filtering down across all the user's attributes.
- The "Filter" icon to the right of the search bar expands the search to include dropdown filters like "Major".
- After submitting the search, user profiles matching the results appear at the bottom, with an expand icon in each to show more details from the user.

# Student Home and Profile page - Adeife Fadahunsi



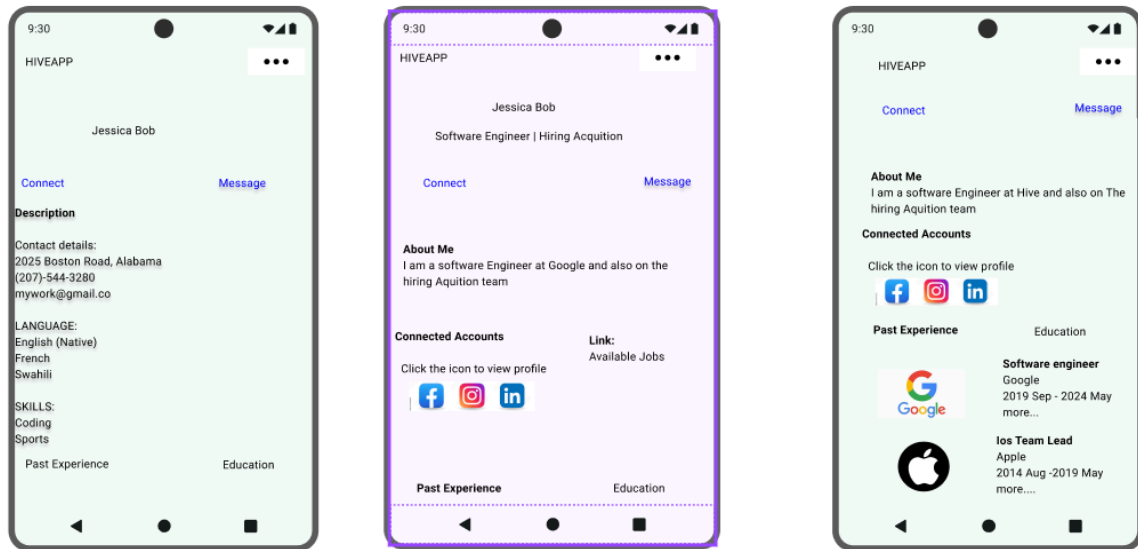
## Screen 1: Home page

- This is the home page of the student's perspective, where they can swipe right and/or click on APPLY to be considered for a job posting by an employer.
- Students can also skip a job posting by swiping left and/or clicking on SKIP not to be considered for a job posting by an employer.
- Students can view job postings from multiple employers, including information on that job posting, such as the title, company, and other factors.

## Screen 2: Profile page

- This is the personal profile page of the student's perspective, where they can view their information, such as their name, email, number, address, and other factors employers consider for a job posting.
- This is also the page where students can update their profile information by making changes to their profile and clicking on the UPDATE PROFILE button. If no changes have been made, the button does nothing.

## Employer Profile Page and Experience - Alisala Mwamba



### Screen 1 and 2: Profile Page

- This is a profile page where all the information regarding the employer has their information to be viewed by other applicants as well as companies as well. This is where you can view the employer's about themselves information, the type of roles that they cover, a way to send a message as well as connecting with them just in case due to already viewing their profile.
- Applicants or students can see all the details that are entailed by the employer along with links for the jobs posted. Applicants are also able to navigate through the employer's social network for more connections as well. The 3 dots at the top gives users the ability to update their profile.

### Screen 3: Past Experience

- This is an extension of the employer's profile, this contains their "past experiences" through their careers as well as a bit of information of what they did at their previous companies in more detail.
- The "education" section entails what background of education they went through in the case of finding jobs through companies.