EASSESSMENT: SOLUTION TO HIRE TALENT

Project Report

Internet & Web System

Utpal Patel

Utpal_patel@student.uml.edu (02006609)

Team Member:

Chinmay Mathakari

Chinmaynandkishor_mathakari@student.uml.edu (02007302)

Table of Content

CHAPTER-1

1. Abstract

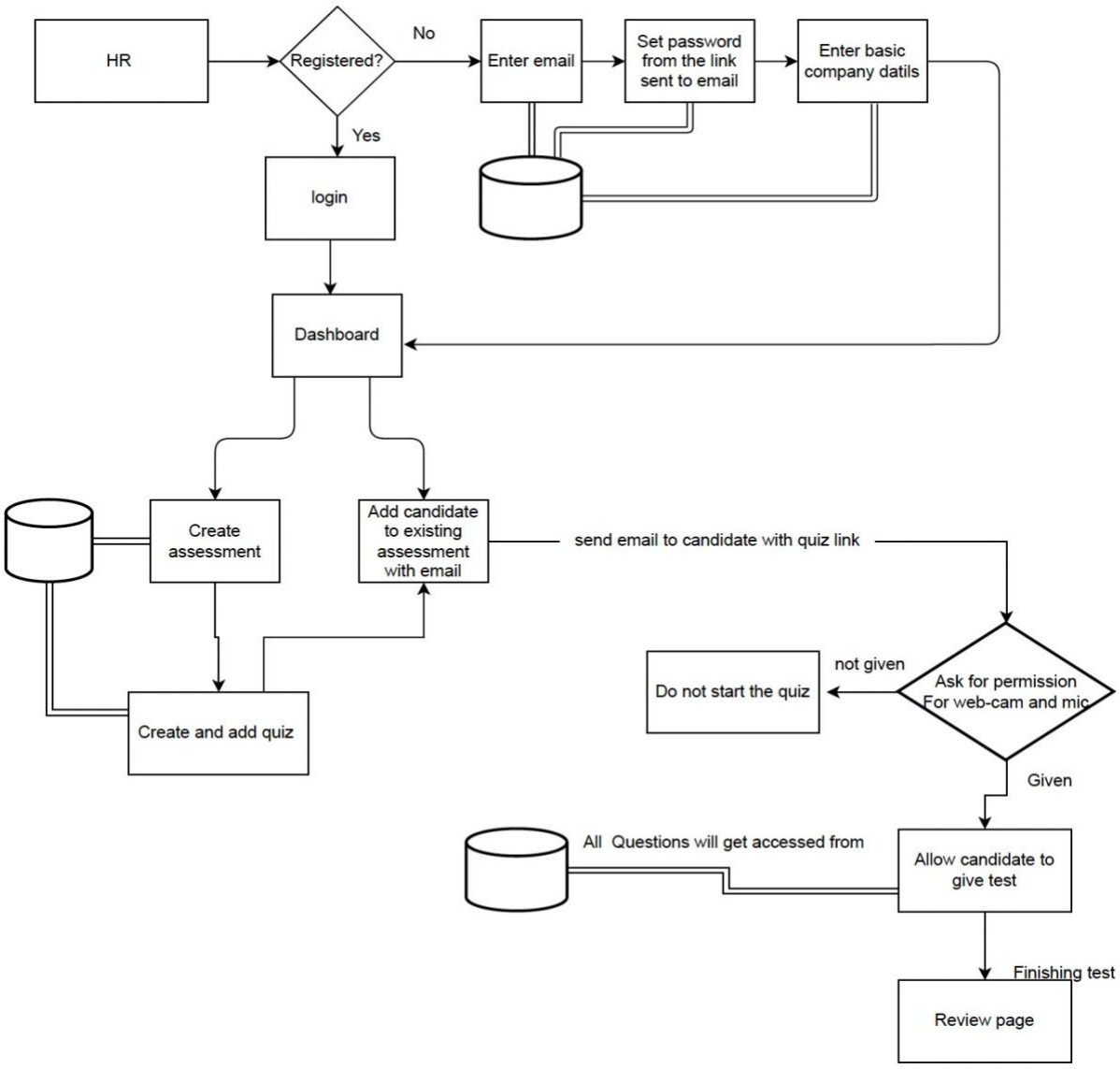
As a result of market competition, many businesses require a platform to hire talent. They will benefit from our web application. Businesses and recruiters can use our platform to employ people and assess them by designing their own questions on various themes. Recruiters can use our web tool to create a test and assess candidates' knowledge and critical thinking skills. A recruiter can register an account here, which will be validated by email via a web page link. The recruiter must include information about himself/herself as well as the company on that site. After creating the profile, the recruiter can design exams that include multiple choice, coding, video, and essay type questions. In this situation, the recruiter will determine the correct answer. For coding questions, we intend to connect an online code editor. In this situation, the recruiter will determine the correct answer. Following the successful creation of an assessment, the recruiter will invite and enter the names and email addresses of candidates for whom the recruiter wishes to perform an assessment. The application will now send a link to a web page to the candidates' email addresses, allowing them to participate in the assessment. The candidate's name, candidate ID, and live image will be required at the start of the assessment. After completing the exam, the candidate's performance will be kept in a database for the employer to view.

CHAPTER-2

2.1 Project work

Our project has mainly two section which is the recruiter and candidate. The recruiter will register in the web application and will be asked to give the company's profile and other information. Now on successful login of recruiter, he/she will be landed on the dashboard page where the recruiter can create new assessment. This page mainly asks the name of assessment and job role and then the recruiter can add the candidate's name, email whom he/she wants to assess. After that, the recruiter will set the questions for the created assessment where the question type mainly divided into multiple choice question, coding question, essay writing, video and multiple select question. While selecting the question module, the recruiter specifies the timing for each question so the candidate will not spend more than the dedicated time to that question. Now, upon successful creation of the assessment, the application will send a weblink to each candidate from where the assessment will be started. Before starting the assessment, the candidate must take his/her picture and allow the camera access to a browser. Also, in the background, the application will be taking the image of the candidate and all the images will be stored into Cloud for the reference of the recruiter.

2.2 Workflow:


HR
Registered?
No
Yes
Enter email
Set password from the link sent to email
Enter basic company datils
login
Dashboard
Create assessment
Add candidate to existing assessment with email
send email to candidate with quiz link
Create and add quiz
Do not start the quiz
not given
Ask for permission For web-cam and mic
Given
All Questions will get accessed from
Allow candidate to give test
Finishing test
Review page

2.3 Division of labor among team members

My work:
I worked on deciding on back-end technology. Firstly, I studied the difference between the implementation of APIs in python and java. I started working on Java and initially, I made APIs to perform simple CRUD (Create, Read, Update, Delete). Gradually, making API that perform JWT authentication and email configuration, I found Java hard. On the other hand, I started trying python to develop APIs as it has many inbuilt libraries and found Flask framework easier to use than Java Spring framework. After deciding the flask framework, I started working on designing APIs related to the project. Major part for me as a newbie to python is to send emails for verification, invitation and also fetching tokens sent by the front-end. I watched videos and referred so many materials provided by python as well as other blocks and completed the work. It was great experience learning python and working on it.

My partner worked on designing webpages and passing payload to post APIs from front end. He also worked on taking screenshots while recording live video and uploading it to Cloudinary.

2.4 Technologies:

Front-End Technologies: React.js

React.js is fast in terms of processing speed. Despite changing every component of HTML page, it will render those components which requires re-rendering so it will directly speed up the rendering process.

When we run the 'npm run build' to start react project, whole source code will be converted to a single chunk of codes where this chunk will run on the server which is the reason why react.js is faster.

Back-End Technologies: Flask framework

Flask is framework of Python which is used to create web applications. It allows us to use ORM (Object Relational Mapping) where we need not to write SQL queries. Moreover, HTTP request handling is easy.

I have used Flask mainly as to get work experience on Python as every big companies are using Python and also I can make a foundation to start working on Machine learning and AI as Python is prerequisite.

Database: MySQL

MySQL is Open-source and fast database. Some of the SQL commands are easier as compared to Oracle DB 11. Moreover, MySQL workbench provides clean UI and text editor. Additionally, we can directly update the data in table given by SELECT query.

CHAPTER-3

3.1 Build/Installation/launch Instruction

o For React

1. For React modules, there are two folders. We need to install node js.
2. You can check node is installed or not by command: node -version
3. If it's installed, then open projects and type the command: npm update
4. After that use 'npm start'.
5. If the node is not installed then install node js from https://nodejs.org/en/download/
6. After successful installation of the node perform steps 2 to 4

o For Python

1. For python, we need to check whether python is installed or not by command: python –version
2. If it is installed then type command: pip install -r requirements.txt. It will download all the lib specified in the file.
3. Type the command: venv\scripts\activate.
4. After that type: flask run.
5. If python is not installed then install it from https://www.python.org/downloads/
6. After installation of python perform steps 2 to 4

3.2 Use of Software

Software used: **Visual Code**

- You can download our project from Github
- Command: Git clone https://github.com/UtpalPatel0055/COMP5130f2022.git
- In the downloaded repository, find the folder called ‘’
- The folder will contain 3 sub folders: backend and 2 folder for front end
- React-frontend folder has functionality for the candidate (to take the assessment)
- React-backend folder has functionality for the recruiter
- Backend folder has all the Flask and REST APIs
- Command to hit the react folders: ‘npm start’ (will run on 3000 port. localhost:3000)
- Command to hit the flask (Name:backend) folder: ‘flask run’

Total users: Recruiter and Candidate

- Recruiter Flow

The recruiter will first sign up into the web application on which he/she will get an email to verify email and set a password.

Upon login, the dashboard will be rendered with the history of previously taken assessment if any.

The dashboard can create new assessment which contains the name of assessment, job role, list of candidates who will give the assessment and the list of question with the dedicated timing per question.

Upon completion of new assessment, the application will send a web link to each candidate who has been added while creating the assessment. The recruiter can later add and remove the candidate later if he/she wants.

- Candidate Flow

The entry point of the candidate is the web link sent by the recruiter via email.

Before the assessment start, the web page asks to capture the image of the candidate and has to allow permission to access the camera.

The front end will click the screen using the camera and candidate has to submit the answers as per the defined time limit of each question by the recruiter.

CHAPTER-4

4. Limitations

- The recruiter cannot delete and edit the assessment as we tried to make APIs where we failed to properly connect with the front end.
- The performance of candidates (answers) is not stored in the recruiter dashboard.
- The recruiter cannot back track while creating the new assessment as we are failed to restore the data from the database if the recruiter does back propagation as per our UI.
- We have attached the code editor, but our system cannot assess the coding answer of the candidate and perform test cases on the given code.

CHAPTER-5

5. Future Work

We can try to correct the design of our database as we have faced numerous errors to connect with react and data parsing between front end and backend. Also, we need to have a UI to store the performance of each candidate so the recruiter can have a web link under the dashboard where he/she can navigate through assessment => candidate list => view results. On the page, the recruiter can see the answers off every single question. It is hard to make a application that check essay and video type question by itself but we might make an algorithm which can automatically give marks to MCQ and multiple selection questions.

CHAPTER-6

References

[1] https://www.geeksforgeeks.org/using-jwt-for-user-authentication-in-flask/

[2] https://dev.to/grahammorby/jwt-auth-in-flask-python-18i4

[3] https://www.youtube.com/watch?v=2NL5BWWA2Qo

[4] https://pythonhosted.org/Flask-Mail/

[5] https://www.geeksforgeeks.org/sending-emails-using-api-in-flask-mail/

[6] https://www.geeksforgeeks.org/python-build-a-rest-api-using-flask/

[7] https://www.geeksforgeeks.org/get-post-requests-using-python/?ref=lbp

[8] https://www.youtube.com/watch?v=Qr4QMBUPxWo

[9] https://www.youtube.com/watch?v=dam0GPOAvVI