



## AITCS

Homepage: <http://publisher.uthm.edu.my/periodicals/index.php/aitcs>  
e-ISSN :2773-5141

# Job Hiring Management System for H.I.S. Management Services

Muhammad Uzair Mohd Faizul<sup>1</sup>, Norhanim Selamat<sup>1\*</sup>,  
Marniza Jamalana<sup>2</sup>

<sup>1</sup>Faculty of Computer Science and Information Technology,  
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400, MALAYSIA

<sup>2</sup>H.I.S. MANAGEMENT SERVICES SDN. BHD.  
Unit 22-16, Q Sentral, 2A, Jalan Stesen Sentral 2, Kuala Lumpur Sentral, 50470  
Wilayah Persekutuan Kuala Lumpur, Malaysia

DOI: <https://doi.org/10.30880/aitcs.2023.04.02.086>

Received 23 June 2023; Accepted 09 November 2023; Available online 30 November 2023

**Abstract:** Job Hiring Management System is a system that will help a company in hiring new employees. Based on the research conducted, the use of Microsoft Word, Microsoft Excel, Gmail and job search engines such as JobStreet are used for hiring. Using multiple software systems can cause delays in the hiring process as information relevant to the hiring process is scattered across different platforms. Moreover, relying solely on JobStreet services for only advertising purposes can be expensive. Therefore, this Job Hiring Management System was developed using web-based technology such as HTML, CSS, JavaScript, Vue.js, Node.js, Express, PostgreSQL and Redis that was designed via Multi-tier Architecture to facilitate access to managers, Human Resources officers and applicants throughout the hiring process. This can help improve the efficiency and effectiveness of Human Resources officers and managers during the hiring process by listing applicant feature for shortlisting and interview purposes. Furthermore, the existence of a centralized system can store all information related to the hiring process as well as reduce the cost of the company's expenses for the smoothness of the hiring process.

**Keywords:** Job Hiring Management System, Human Resources, Web Technology, Throwaway Prototyping, Multi-tier Architecture

## 1. Introduction

This study involves one of the companies related to business management (HIS Management Services) which involves a travel company (HIS Travel), which is also its own parent company. Currently, hiring process is through the JobStreet application. However, the problem that arises is when Human Resources officers need to manage data and information outside the JobStreet application such as in Microsoft Word, Microsoft Excel and Gmail regarding applicants for any vacant positions.

Using more than one software will slow down the hiring process, which sometimes takes a short time to fill important job vacancies. In addition, the use of more than one software also causes important

information related to the hiring process to be in different places. This may cause Human Resources officers to overlook or forget about it.

So far, the JobStreet service is used for advertising purposes only to find potential employees for the company. The cost invested is quite high at the moment because this company is also affected by the COVID-19 Pandemic which slows down the tourism industry [1]. Now, the post-Covid-19 economy has just started to improve the tourism sector after the relaxation of some restrictions in most countries [2]. This also includes countries that have HIS Travel branches.

The main objective of this project is to analyze and design a Job Hiring Management System using an object-oriented approach, develop a Job Hiring Management System based on the Web Technology approach and perform alpha testing of the developed system. The main scope highlighted is the process of new recruitment notices, advertisements and applying for vacancies, managing profiles, managing interview sessions, informing applicants and managing company accounts.

## **2. Literature Review**

### **2.1. Job Hiring Management System**

Job Hiring Management System is a system that will help an organization or company in hiring new employees. Through this system, the management of the new hire process can be completed more easily because there is a centralized platform that will help store all the information related to it.

According to Indeed [3], a fairly popular global job vacancy advertising portal, there are seven (7) basic steps in the hiring process, which is identifying company needs, making hiring plans, drafting job and job details, advertising vacancies, analyzing candidates, conduct interview sessions and close the hiring process.

### **2.2. The technology used**

This system is a system based on web technology. There are four sections for the development of this system namely Front-End, Back-End, Database and Cloud Infrastructure. On the Front-End side, the technologies used are HTML, CSS, JavaScript, TailwindCSS, Vue.js and Capacitor. In addition, on the Back-End side, the technologies used are JavaScript, Node.js and Express. Furthermore, on the Database side, the technologies used are PostgreSQL and Redis. Finally, in the Cloud Infrastructure section (Platform as a Service), the technology used is Render Cloud Services. Basically, the sections of these four parts is based on the chosen system architecture which is Multi-tier Architecture.

### **2.3. Comparison between different systems**

There are several systems in place today that are closely related to the hiring process. These systems are studied and analyzed for each usage. Some advantages and limitations have been identified in the existing systems. Therefore, these limitations can be considered and addressed through the features and modules introduced in this proposed system. Moreover, the advantages of these systems will be used as the most important inspiration in developing the proposed system. Among the existing systems are JobStreet, Indeed, Personio and Freshteam. Table 1 shows the results of the system comparison.

**Table 1: Comparison between different systems**

Aspect / System	JobStreet	Indeed	Personio	Freshteam	<i>Proposed System</i>
The technology used	Web, Android	Web, Android	Web, Android	Web, Android	Web, Android
Register User and Login	Yes	Yes	Yes	Yes	Yes
Feature of processing recruitment notices	No	No	Yes	Yes	Yes
Feature of advertising job vacancies	Yes	Yes	No	Integration with other systems such as LinkedIn needs to be done	Yes
Feature of viewing list of applicants	Yes	Yes	Applicant listing done by manually	Applicant listing based on integration with other systems such as LinkedIn	Yes
Feature of managing applicant profile	Applicant profile managed by the applicant himself/herself	Applicant profile managed by the applicant himself/herself	Applicant profile managed by HR officer.	Applicant profile managed by HR officer	Applicant profile managed by the applicant himself/herself
Feature of listing interview candidates	No	No	Yes	Yes	Yes
Feature of informing candidates for	No	No	No	No	Yes

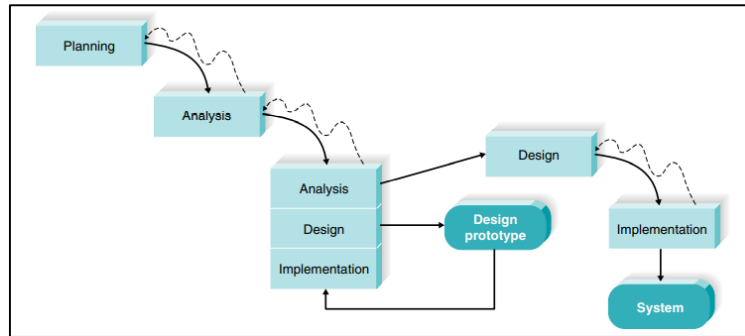
Aspect / System	JobStreet	Indeed	Personio	Freshteam	Proposed System
interview or hired					
Specifically for advertising job vacancies	Yes	Yes	No	No	No
Specifically for finding new employee	No	No	Yes	Yes	No
Covers the entire of hiring process	No	No	No	No	Yes

Based on Table 1, JobStreet and Indeed are web and Android-based platforms offering user registration and login capabilities. However, they lack specific features for processing recruitment notices. On the other hand, Personio, Freshteam, and the proposed system provide features for processing recruitment notices, as well as managing applicant profiles and listing interview candidates. The proposed system stands out with its ability to inform candidates about interview results and hiring decisions. It also covers the entire hiring process, integrating various functionalities such as applicant profile management, interview listing, and candidate communication. While JobStreet and Indeed focus more on advertising job vacancies, the proposed system offers a comprehensive approach to streamline the hiring process.

### 3. Methodology

#### 3.1. Throwaway Prototyping Model

The system development model as the methodology for this project is using the Throwaway Prototyping model. The planning phase is about planning the structure for the project. The analysis phase is to obtain information and requirements about the To-Be System from the stakeholders involved with it, documents or As-Is System [4]. The design phase is about designing the components, architecture and interface of the system to be developed. The implementation phase is the phase of developing and testing the system. Furthermore, this Throwaway Prototyping will produce a design prototype, which after gathering information and requirements, a design prototype will be made to further improve the understanding of features or components in the system. After that, the analysis phase will resume to check and validate the design prototype if any problems arise or not. Then, the design prototype will be used as a guide to design the architecture and interface of the entire system and as a reference to develop the actual system. For further information, this project will produce two (2) design prototypes before developing the actual system. Figure 1 shows the Throwaway Prototyping model used as a system development model.



**Figure 1: Throwaway Prototyping Model [5]**

### 3.2. System Development Workflow

The workflow needs to be done to ensure that each phase has appropriate and accurate activity, and the allocated time can be used optimally. Table 2 below explains each task that needs to be settled and completed.

**Table 2: System Development Workflow**

Phase	Task(s)
Planning	<ul style="list-style-type: none"> <li>• Make project planning based on the selected project methodology.</li> <li>• Coordinate the activity and duration for project timeline.</li> <li>• Discuss with stakeholders to obtain information related to the project via interviews and documents.</li> </ul>
Analysis	<ul style="list-style-type: none"> <li>• Collect all important information from the interview process and documents provided.</li> <li>• Analyze the main issues from the requirements elicitation.</li> <li>• Build a UML diagram based on the requirements elicitation.</li> </ul>
Creating Prototype	<p>Analysis</p> <ul style="list-style-type: none"> <li>• Gather all the important information from the requirements elicitation.</li> <li>• Refers to UML diagrams to develop a feature/module.</li> </ul> <p>Design</p> <ul style="list-style-type: none"> <li>• Produce interface sketches for features/modules to be developed.</li> <li>• Build design prototype.</li> </ul> <p>Implementation</p> <ul style="list-style-type: none"> <li>• Test the design prototype.</li> <li>• Validate the design prototype based on requirements elicitation.</li> </ul>
Design	<ul style="list-style-type: none"> <li>• Refers again the architectural design as well as interface designs that have been created.</li> <li>• Choose an appropriate system architecture design and interface design.</li> <li>• Create a database design for data structuring.</li> </ul>
Implementation	<ul style="list-style-type: none"> <li>• Organize and perform programming tasks.</li> </ul>

Phase	Task(s)
	<ul style="list-style-type: none"> <li>• Test the system development using unit testing and end-to-end testing.</li> <li>• Conduct a performance analysis auditing of the system.</li> </ul>
Testing	<ul style="list-style-type: none"> <li>• Conduct alpha testing after system deployment is complete.</li> </ul>

### 3.3. Functional Requirements

Functional requirements describe how the system should operate. Meaning, each module or feature that is introduced must have certain requirements for how the system implements the module. That is why, this functional requirement is taken. Table 3 shows the functional requirements for the Job Hiring Management System.

**Table 3: Functional Requirements**

No..	Module	Description
1.	Create new company account	Before conducting any recruitment tasks in the system, a new company account needs to be created.
2.	Register new user	Users such as HR officers, managers, applicants, and guest need to register as new users if they do not have an account.
3.	Log in	Any recruitment application and employee hiring management tasks require logging in first.
4.	Process a new recruitment notice	Before hiring new employees, a recruitment notice needs to be issued by the department manager who intends to hire new employees.
5.	Advertise and apply a job	The HR officer can advertise the vacant position. Applicants can apply for the job based on the eligibility criteria.
6.	Manage profile	HR officers, managers, applicants, account administrators, and guest who have logged in can add detailed information to their profiles or edit profile information.
7.	Manage interview sessions	HR officers shortlist candidates from the list of applicants. The interview sessions will be communicated by the HR officer to the manager or account administrator and guest (if involved).
8.	Notify applicants	The HR Officer will notify successful applicants about the interview and selection as a new employee of the company.
9.	Manage company account	All data and information can be managed by the account administrator.

### 3.4. Non-Functional Requirements

Non-functional requirements can be defined as specific features or behaviours that a system should have. According to Dennis, A., Wixom, B. H., & Roth, R. [5] non-functional requirements are the main focus in the design phase when decisions are made about the user interface, hardware and software and

the basic architecture of the system. Table 4 explains the non-functional requirements of the Job Hiring Management System.

**Table 4: Non-Functional Requirements**

No.	Type	Description
1.	Operation	<ul style="list-style-type: none"> <li>The system is web-based and can be accessed and used using modern browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, Opera, Brave, and others. However, Internet Explorer browser, which has ended its support by Microsoft, is not suitable for accessing this system.</li> <li>For systems based on Android, the system only supports the use of Android version 5.1 and above.</li> </ul>
2.	Usability	<ul style="list-style-type: none"> <li>The system is designed to be compatible with devices of various screen sizes such as smartphones, laptops, and monitors using Responsive Design techniques.</li> </ul>
3.	Accessibility	<ul style="list-style-type: none"> <li>The system requires an internet connection to be used. The system will not function properly if there is no internet connection while the user interacts with it.</li> </ul>
4.	Security	<ul style="list-style-type: none"> <li>Users need to log in to the system using their email and password, based on their respective roles such as applicants, HR officers, managers, account administrator and guests.</li> <li>If a user is inactive for seven (7) days after logging in, the user needs to log in again to access the system.</li> <li>Each user with different roles has a different interface, and they cannot access each other's interfaces. For example, applicants cannot access the HR officer's interface.</li> <li>User passwords must be at least eight (8) characters long, including a mix of uppercase letters, lowercase letters, and numbers.</li> </ul>
5.	Integrity	<ul style="list-style-type: none"> <li>User passwords are not stored in plain text in the database. Hashing and salting techniques are used.</li> <li>User messages are not stored in plain text in the database. Symmetric encryption techniques are used.</li> </ul>
6.	Maintainability	<ul style="list-style-type: none"> <li>The system is deployed through Cloud Services (Render Cloud Services), and the source code is stored in a Git repository (GitHub). Therefore, any modifications or code changes will not affect the operations and data of the existing running system.</li> </ul>

No.	Type	Description
7.	Performance	<ul style="list-style-type: none"> <li>The system should have a response time of less than seven (7) seconds if the internet connection is good.</li> <li>The system's performance will be audited using Google Lighthouse. Therefore, the targeted score should exceed 70.</li> </ul>

### 3.5. Use Case Diagram

According to Unhelkar B. [6], the use case diagram is a system requirements model roughly. This diagram are used to illustrate use cases, their corresponding actors and their interactions. Visual models of use cases facilitate understanding of business processes and help in communicating with stakeholders. The specifications shown in the use case diagram form the most important content in the requirements modeling of the developed system. Figure 2 shows the use case diagram for the Job Hiring Management System.

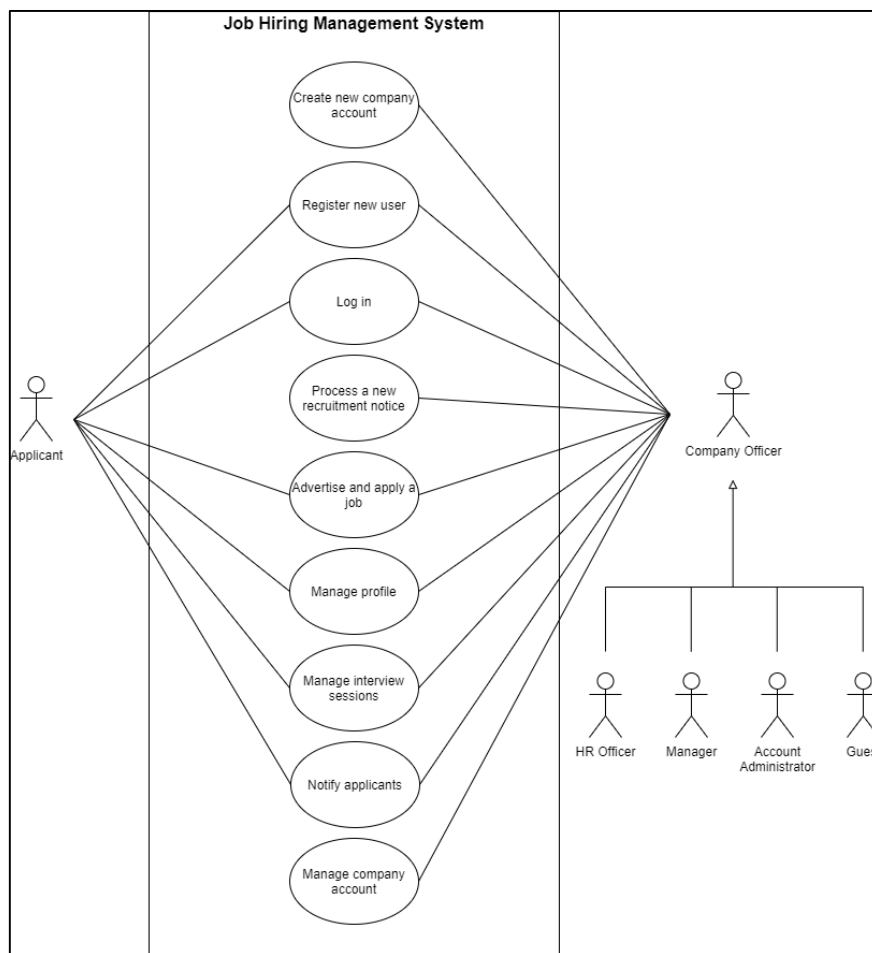


Figure 2: Use Case Diagram

Based on Figure 2, there are two (2) main type of user, which are applicant and company officer. The company officer represents four (4) type which are HR officer, manager, account administrator and guest. Besides, there are nine (9) use cases for this system. Among them are create a new company, register new user, log in, process a new recruitment notice, advertise and apply a job, manage profile, manage interview sessions, notify applicants and manage company account.

### 3.6. Class Diagram

After the collection of information to determine the logic and rules for the developed system, the determination of the data involved is also implemented. In any system though, data is very important to provide or store it for users. Thus, the importance of this class diagram will show how the data is connected to each other. Figure 3 shows the class diagram.

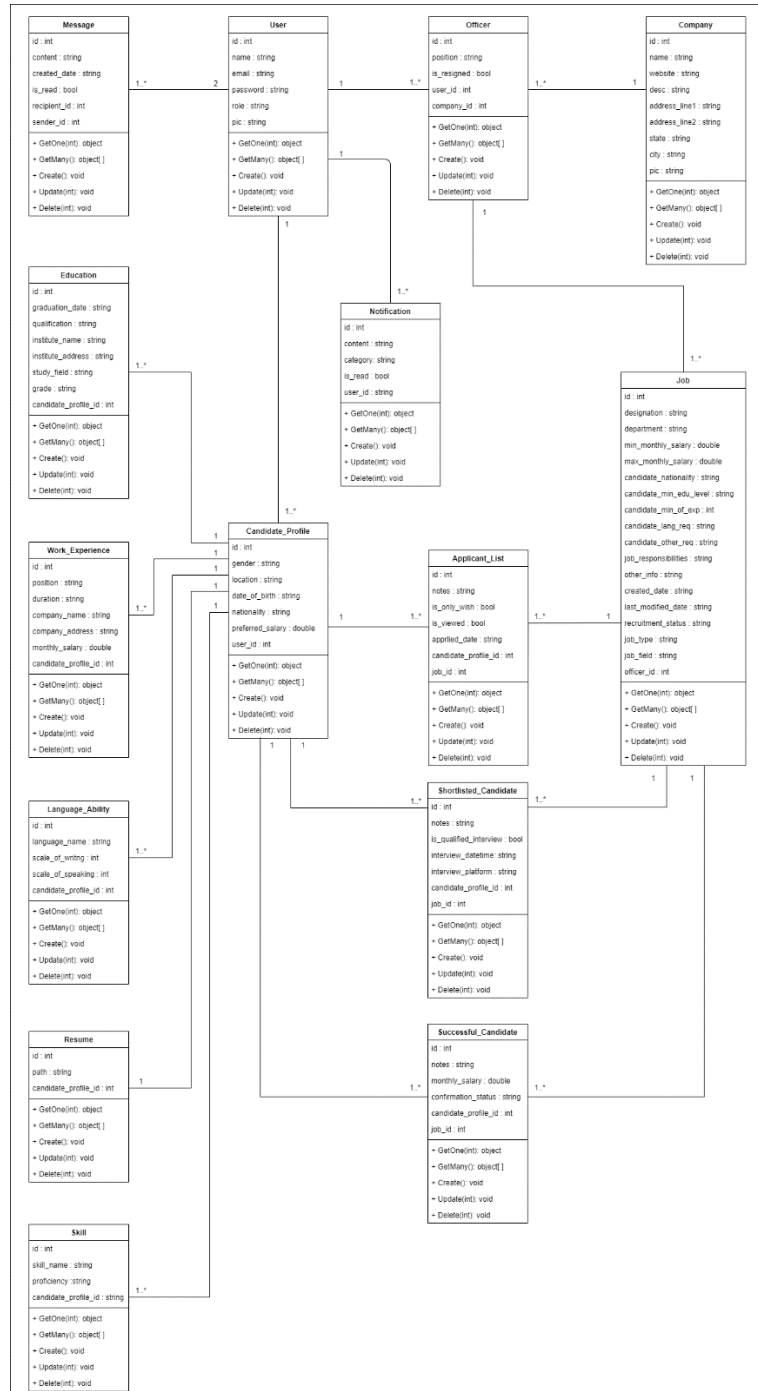


Figure 3: Class Diagram

Based on Figure 3, there are 16 classes created which are user, message, notification, officer, company, job, trash bin, applicant list, shortlisted candidate, successful candidate, candidate profile, education, work experience, language ability, resume and skill. This class diagram was normalized to 3<sup>rd</sup> Normal Form (3NF) which the table is very specifically and connected via foreign key. Other than

that, there is no many-to-many relationship between classes which can make data fetching easier. Overall, this class diagram represents the table that was created in the database.

### 3.7. System Architecture

The system architecture chosen for this Job Hiring Management System is Multi-tier Architecture. Named multi-tier or n-tier because in the architecture of this system, there are several layers or components that have their own functions [7]. Each component also has a relationship with each other. The CDN Front-End component is used to host the Front-End part which is also the system interface and will be seen by the user in the web browser. The Client component of the mobile application will also show the system interface. The Back-End Server component controls the algorithm and logic for the data and business involved in the system. The Database Server component uses PostgreSQL to store data, the In-Memory Data Store component to store important data in memory because the process of retrieving it is faster than storing it in PostgreSQL and the Persistent File Storage component to store files such as resumes in PDF form and profile pictures user in PNG or JPG format. Figure 4 shows how the components in the Multi-tier Architecture are formed.

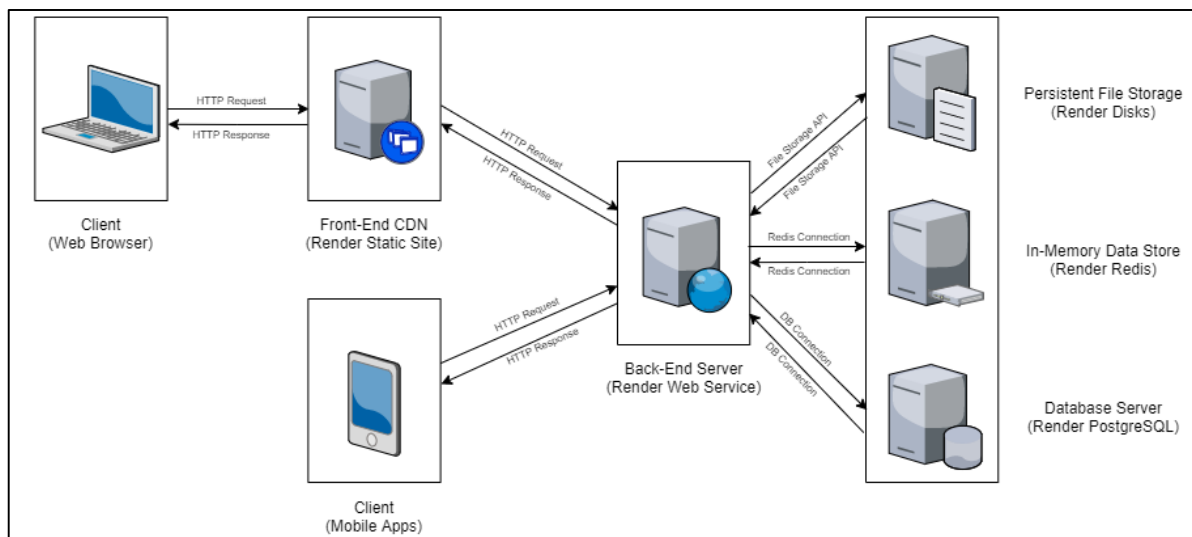


Figure 4: Architecture for Job Hiring Management System

## 4. Result and Discussion

### 4.1. Implementation

The implementation phase in software development is an important stage where the planned software will be developed and deployed. It is the phase where the design, requirements and specifications gathered during the previous stage are transformed into an actual software product. The main goal of the implementation phase is to build a software system that meets the identified requirements and provides the intended functionality. Figure 5-8 shows the interface for main module in the system, and Figure 9-12 shows the code segment for respective interface.

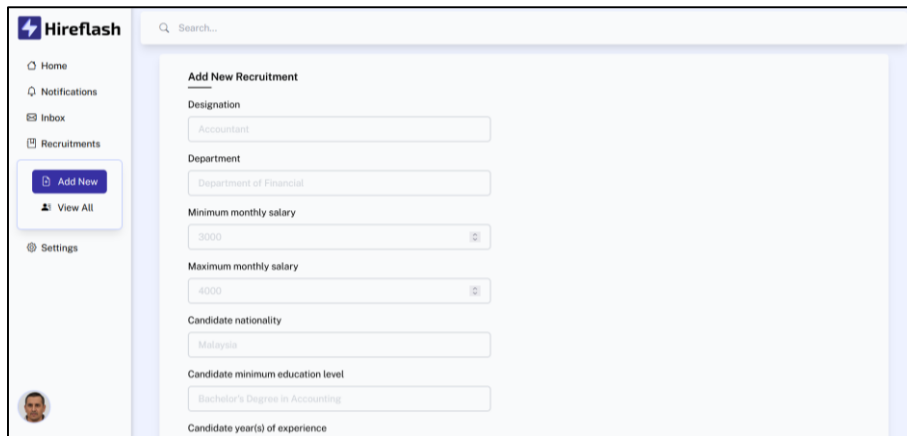


Figure 5: Interface for creating a new recruitment

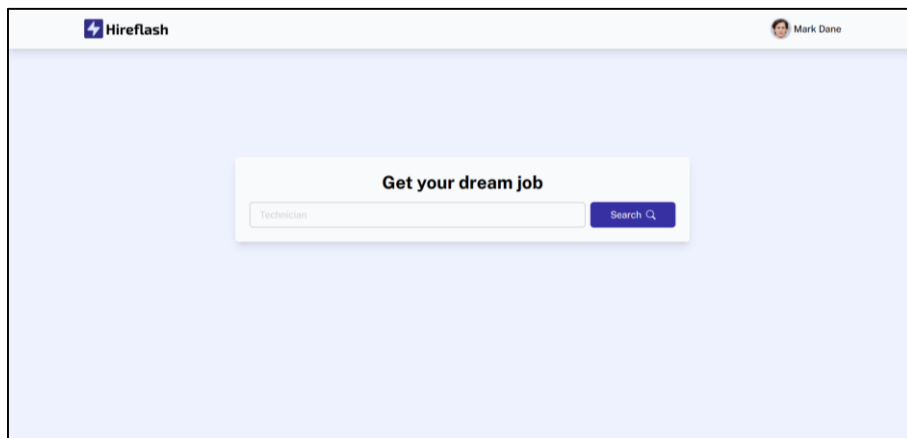


Figure 6: Interface for searching a job

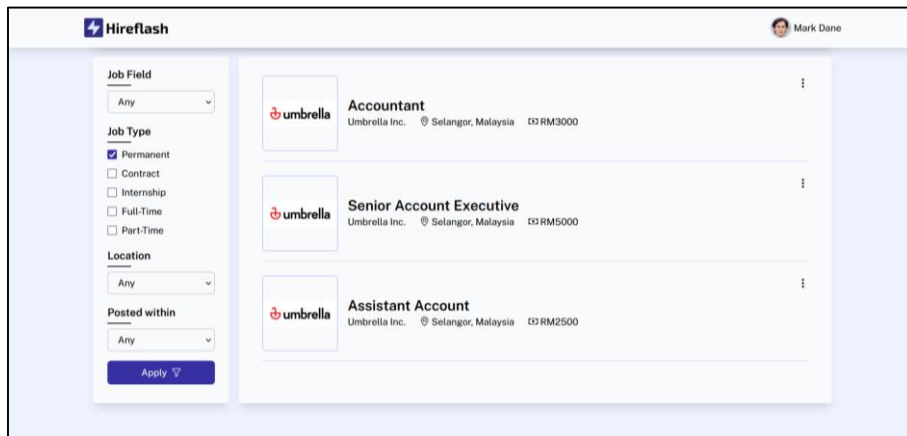


Figure 7: Interface for job result from search

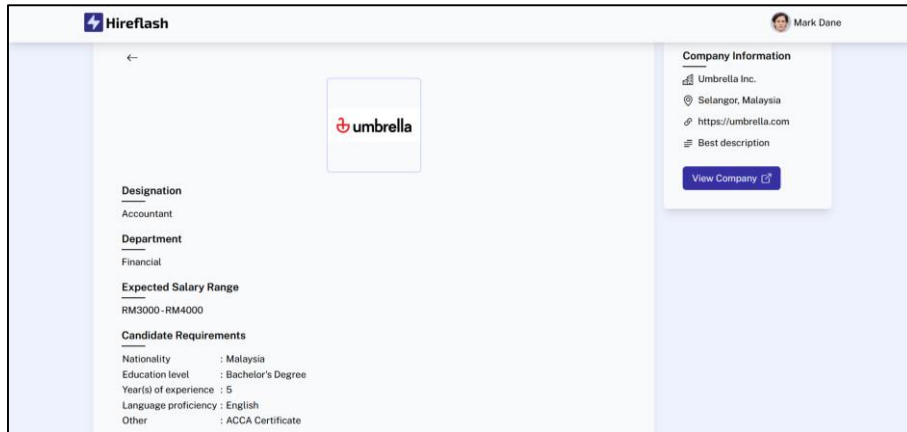


Figure 8: Interface for viewing a job detail

```

1  const response = await fetch(baseEndpoint + "/job", {
2    method: "POST",
3    mode: "cors",
4    headers: { "Content-Type": "application/json" },
5    credentials: "same-origin",
6    body: JSON.stringify({
7      designation: designation.value,
8      department: department.value,
9      min_monthly_salary: minsalary.value,
10     max_monthly_salary: maxsalary.value,
11     candidate_nationality: nationality.value,
12     candidate_min_edu_level: eduLevel.value,
13     candidate_min_of_exp: experience.value,
14     candidate_lang_req: lang.value,
15     candidate_other_req: otherReq.value,
16     job_responsibilities: jobRespons.value,
17     other_info: otherInfo.value,
18     created_date: dayjs().format("YYYY-MM-DD"),
19     last_modified_date: dayjs().format("YYYY-MM-DD"),
20     recruitment_status: status.value,
21     job_type: jobType.value,
22     job_field: jobField.value,
23     officer_id: officer.value.id,
24   }),
25 });

```

Figure 9: Code segment for creating a new recruitment

```

1  const jobList = await prisma.job.findMany({
2    where: {
3      recruitment_status: "Advertised",
4    },
5    include: {
6      company: true,
7    },
8  });
9
10 const options = {
11   includeScore: true,
12   keys: [
13     "designation",
14     "job_responsibilities",
15     "other_info",
16     "job_field",
17   ],
18 };
19
20 const fuse = new Fuse(jobList, options);
21
22 let list = fuse.search(query);

```

Figure 10: Code segment for searching a job

```

1  const response = await fetch(
2    baseEndpoint + "/job/" + router.currentRoute.value.params.jobId,
3    {
4      method: "GET",
5      mode: "cors",
6      headers: { "Content-Type": "application/json" },
7      credentials: "same-origin",
8      // body: JSON.stringify({}),
9    }
10 );

```

Figure 12: Code segment for viewing a job detail

#### 4.2. Deployment

Deployment refers to the process of making a software application or system, available and operational for its intended users. It involves the installation, configuration and setup of an application or system on a target environment, be it a server, cloud platform or user device. Before the deployment process is done, the source code for this system will be stored in GitHub to facilitate the process of source control and version control, backup for remote storage and software project management. After that, the source code of the Frontend and Backend parts of the developed system will be placed in Render. Figure 13 and 14 shows the GitHub repository for frontend and backend source code.

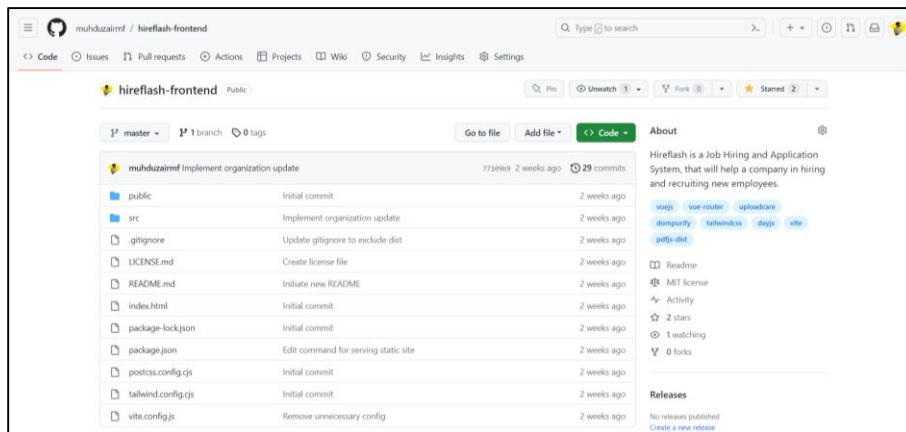


Figure 13: GitHub repository for frontend source code

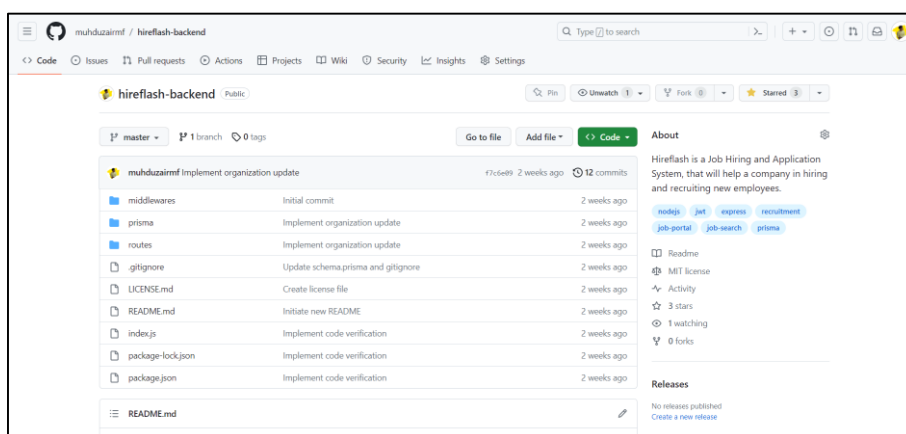


Figure 14: GitHub repository for backend source code

Deployment for the Frontend and Backend parts will use two different types of services in Render. Placement for the Frontend part will use the Static Site feature and Placement for the Backend part will use the Web Services feature. Data storage and file storage are also located in Render using the Render PostgreSQL, Render Redis and Render Disks features.

### 4.3. Functional Testing

Functional testing is a type of software testing that focuses on verifying the functional requirements and specifications of a software application or system [8]. It involves evaluating the behaviour and functionality of an application against the expected results to ensure that it fulfills its intended purpose and can function properly. Table 5 shows the summary of the test cases for this system functional testing.

Table 5: Summary of the test cases for this system functional testing

Test ID	Requirement ID	Description	Result
TC01-01	FR01-01	The system needs to allow account administrators to access the get started page.	PASS
TC01-05	FR01-05	The system needs to confirm the verification code received.	PASS
TC01-05	FR01-06	Account administrators can enter a name, password and password confirmation.	PASS
TC01-07	FR01-07	The system needs to verify the received company details.	PASS
TC02-03	FR02-03	The system needs to verify the name, email, password and password confirmation received.	PASS
TC03-03	FR03-03	The system needs to confirm the received email and password.	PASS
TC04-02	FR04-02	Managers can fill in the details required to create a new recruitment notice.	PASS
TC04-05	FR04-05	Managers can update the status of new recruitment notices	PASS
TC05-03	FR05-03	The system should display a list of advertised job vacancies to applicants.	PASS
TC05-04	FR05-04	Applicants can search for job vacancies according to specific filters or text searches.	PASS
TC06-02	FR06-02	Users can update profiles.	PASS
TC07-01	FR07-01	The system displays a list of applicants for advertised job vacancies.	PASS
TC07-02	FR07-02	HR Officers can view detailed information for each applicant.	PASS
TC08-04	FR08-04	HR Officer can send a message to the applicant for any job entry matters.	PASS
TC08-05	FR08-05	HR Officer can update into the system that job hiring is on hold.	PASS

Test ID	Requirement ID	Description	Result
TC09-02	FR09-02	Account administrators can update company information.	PASS

#### 4.4. User Acceptance Testing

User Acceptance Testing is the phase of software testing that focuses on verifying whether the software system meets the needs and expectations of end users [8]. It involves testing software in real-world user environments to ensure usability, functionality and overall satisfaction. This testing is usually done by end users or representatives of the targeted user base, rather than the system development team or the Quality Assurance team. Figure 15 shows the chart of the result from the User Acceptance Testing Form specifically for search and apply the job feature for applicant.

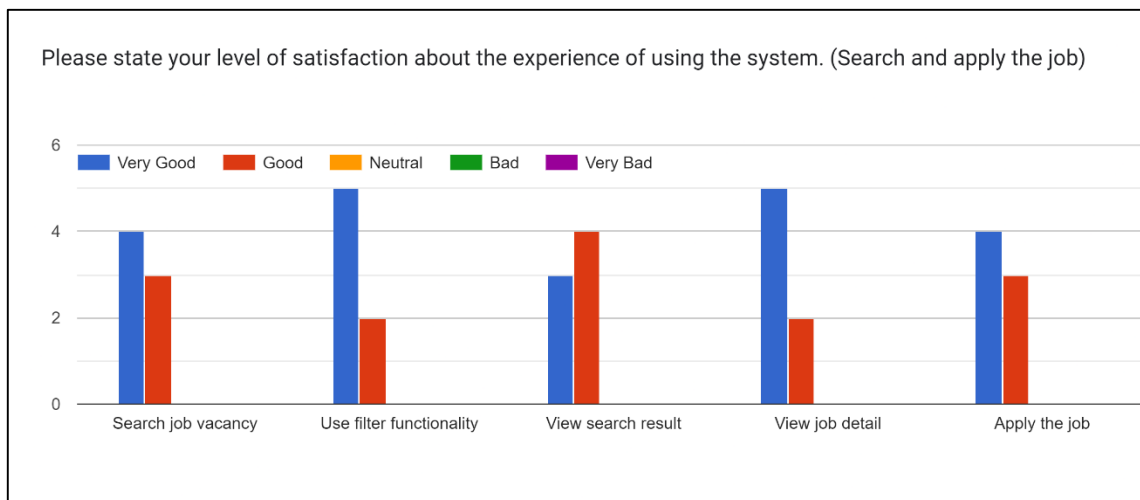


Figure 15: Chart of the result specifically for search and apply the job feature.

Based on Figure 15, there are seven (7) initial users that was invited to join user acceptance testing for this system. As mentioned before, these users are applicant type of the user in the system. This chart is the result of the user acceptance testing where they evaluate the search and apply job feature, which this is the main feature of the system.

## 5. Conclusion

In summary, this Job Hiring Management System was developed based on the main objective highlighted in the project which is to analyze and design a Job Hiring Management System using an object-oriented approach, develop a Job Hiring Management System based on the Web Technology approach and perform alpha testing of the developed system. The developed system also prioritizes nine modules for four users, namely applicants, HR officer account administrators, managers and guests. Thus, the process of analysis, design, implementation and testing is based on the modules. Through the results of user acceptance testing targeting the four users, this system was found to be suitable for use by the targeted users. Other than that, there are a few advantages of this system. For example, this system covers the entire hiring process, the information regarding the opening of vacancies and applications for vacancies can be stored safely for future purposes, the information regarding past applicants can be stored safely for future purposes, searching and filtering feature capability to find job vacancies specifically and exclusively, extra authentication layer for creating new organization or forgot password and extra warning capability for deleting crucial data such as user, organization or job recruitment. However, this system also several limitations such as limitations on messaging feature, applicant may apply for any job vacancy without having complete profile information, access limits for each officers role, no mobile application for iOS operating system and no notification for mobile

application. Although this system still has some limitations, however, this problem can definitely be improved in the future with more efforts and the improvement of additional functions into the system will make this system better accepted by the targeted users.

### **Acknowledgement**

The author would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for the support and encouragement throughout the process of conducting this study.

### **References**

- [1] Mohamad Ali, N. A., Ab Wahab, S. A., & Pardi, D. F. (2021). *COVID-19 dan sektor pelancongan*. RONA TINTA.
- [2] Assaf, A., & Scuderi, R. (2020). *COVID-19 and the recovery of the tourism industry*. *Tourism Economics*, 26(5), 731-733.
- [3] Indeed Editorial Team. (2021). *What Is Recruitment in HR?: Definition and 7-Step Process*. Dicapai pada 24 November 2022 daripada <https://www.indeed.com/career-advice/career-development/recruitment-in-hr>
- [4] Pohl, K. (2016). *Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam-foundation level-IREB compliant*. Rocky Nook, Inc.
- [5] Dennis, A., Wixom, B. H., & Roth, R. M., (2018) *Systems analysis and design*. 5th ed. New Jersey: John Wiley and Sons.
- [6] Unhelkar, B. (2018). *Software engineering with UML*. CRC Press.
- [7] Alseelawi, N. S., Adnan, E. K., Hazim, H. T., Alrikabi, H., & Nasser, K. (2020). *Design and implementation of an e-learning platform using N-TIER architecture*.
- [8] Spillner, A., & Linz, T. (2021). *Software Testing Foundations: A Study Guide for the Certified Tester Exam-Foundation Level-ISTQB® Compliant*. dpunkt. verlag.