HIRESPHERE

Submitted by

SARANG S	:	33221825052
SIDHARTH SIVAN	:	33221825057
SOORYA NARAYAN R S	:	33221825058
VIKAS S	:	33221825063

Guided by

Mrs. Deepthi Rani S.S

Assistant Professor

CHRIST NAGAR COLLEGE
MARANALLOOR, TRIVANDRUM
A CMI Educational Institution | Affiliated to the University of Kerala

PROJECT REPORT

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF COMPUTER APPLICATION DEGREE OF UNIVERSITY OF KERALA

2023

1.

CHRIST NAGAR COLLEGE

MARANALLOOR, TRIVANDRUM

A CMI Educational Institution | Affiliated to the University of Kerala

CERTIFICATE

CERTIFIED THAT THIS REPORT TITLED **HIRESPHERE** IS A BONAFIDE RECORD OF THE MINOR PROJECT WORK DONE BY **SARANG S (33221825052), SIDHARTH SIVAN (33221825057), SOORYA NARAYAN R S(33221825058) AND VIKAS S (33221825063)** UNDER OUR SUPERVISION AND GUIDANCE, AT THE DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS CHRIST NAGAR COLLEGE TOWARDS THE PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF **BACHELOR OF COMPUTER** OF THE UNIVERSITY OF KERALA.

HEAD OF THE DEPARTMENT INTERNAL GUIDE

EXTERNAL EXAMINER

1. …………………………………

2. ………………………………….

ACKNOWLEDGEMENTS

We are using this opportunity to express our sincere gratitude to everyone who supported us throughout the course of this BCA project. We first like to thank the almighty for showering his blessings upon us.

We would like to thank our Manager, **Rev.Fr. Cyriac Madathil CMI** for the encouragement provided.

Then we would like to thank our Principal, **Dr. Jolly Jacob** for the support and encouragement provided.

We express our sincere gratitude towards our Head of the Department of Computer Applications, **Mrs. Libi Kurian** for providing an opportunity for undertaking this project.

We would also like to thank our project guide, **Mrs. Deepthi Rani S S,** Assistant Professor, for all the support given by her throughout the completion of the project.

Last but not least we would like to express our profound thanks to all our friends for their support. We thank all the above-mentioned people for their aspiring guidance, invaluably constructive criticism and friendly advice during the project work. We are sincerely grateful to them for sharing their truthful and illuminating views on a number of issues related to the project.

WITH GRATITUDE,

Sarang S

Sidharth Sivan

Soorya Narayan R S

Vikas S

ABSTRACT

Our project, the Hiresphere, is a comprehensive platform designed to connect job seekers and employers in a user-friendly and efficient manner. In today's digital age, the traditional methods of job searching and recruitment have become outdated and time-consuming. Our software aims to streamline the process by providing an online platform where job seekers can create profiles, upload resumes, and browse through various job listings. Employers, on the other hand, can post job openings, search for suitable candidates, and manage the entire hiring process. The system also incorporates features such as advanced search filters, automated notifications, and secure communication channels to enhance the overall user experience. With the Hiresphere, we aim to revolutionize the way job seekers find employment opportunities and employers identify qualified candidates, ultimately bridging the gap between job seekers and employers in the digital era.

CONTENTS

1. INTRODUCTION……………………………………………………………1

1.1 ABOUT THE PROJECT 2

2. SYSTEM ANALYSIS 3

2.1 EXISTING SYSTEM 5

2.1.1 Limitations of Existing System 5

2.2 PROPOSED SYSTEM 7

2.2.1 Advantages of Proposed System 8

2.2.2 Features of Proposed System 9

2.3 REQUIREMENT SPECIFICATIONS 11

2.3.1 Hardware Requirements 11

2.3.2 Software Requirements 11

2.3.4 Functional Requirements 12

2.3.5 Non-Functional Requirements 15

2.4 FEASIBILITY STUDY 16

2.4.1 Economic feasibility 16

2.4.2 Operational Feasibility 16

2.4.3 Technical feasibility 17

3. SYSTEM DESIGN 18

3.1 DESIGN OF SUBSYSTEM 19

3.2 USER INTERFACE DESIGN 20

3.2.1 Input Design of User 20

3.2.2 Output Design of User 21

3.3 DATABASE DESIGN 22

3.3.1 Table Design 25
3.3.2 Data Dictionary 27
3.4 MODELLING 28
3.4.1 Data flow diagram (DFD) 28
3.4.2 Entity relationship model 31
3.5 ARCHITECTURAL DESIGN 34
3.5.1 Structure Chart 34
3.6 PROCEDURAL DESIGN 35
3.6.1 Flow Chart 35
4. PROCESSING ENVIRONMENT 40
4.1 HARDWARE SPECIFICATIONS 41
4.2 SOFTWARE SPECIFICATION 43
5. CODING AND IMPLEMENTATION 45
5.1 CODING 46
5.2 TESTING 57
5.2.1 Unit Testing 57
5.2.2 Integration Testing 58
5.2.3 System Testing 58
6. SECURITY, BACKUP AND RECOVERY MECHANISM 60
7. FUTURE ENHANCEMENT 62
8. SOFTWARE MAINTENANCE 64
9. CONCLUSION 67
10. APPENDIX 69
GANTT CHART 70

SCREENSHOTS 71

MENU TREE.. 74

MEETING MINUTES 75

11. BIBLIOGRAPHY 81

INTRODUCTION

1.1 ABOUT THE PROJECT

Our Hiresphere revolutionizes job searching and recruitment by connecting job seekers and employers in a user-friendly platform. Job seekers can create profiles, upload resumes, and browse job listings with advanced search filters. Employers can post openings, search for candidates, and manage the hiring process. The portal offers automated n otifications, secure communication channels, and a seamless user experience. Job seekers can track applications and access career resources, while employers can filter candidates and schedule interviews. With a focus on user-friendliness and robust security, our project aims to transform how people find job opportunities and organizations discover talent. By leveraging technology, we drive efficient connections in the dynamic job market, fostering growth and success for all stakeholders.

SYSTEM ANALYSIS

INTRODUCTION TO SYSTEM ANALYSIS

The aim of our project, Hiresphere, is to create a user-friendly platform for job seekers and companies. Our objective is to provide a seamless experience for both parties, allowing companies to easily access job seeker details and job seekers to apply for specific jobs in desired companies. Companies can securely log in to the system using valid credentials. Previously, job seekers' achievements were manually recorded, making it difficult to access and store the information. This manual process was vulnerable to physical and climatic factors, with no alternative means of retrieval. By implementing our system, the need for paper entries is eliminated, reducing human intervention and associated errors. Companies can log in via their desktops and track relevant details about each job seeker, including personal information and educational qualifications. Hiresphere aims to streamline the job search and hiring process, enhancing efficiency and improving the overall experience for job seekers and companies.

2.1 EXISTING SYSTEM

In the existing system, job seekers and companies face limitations in accessing and managing job-related information. Job seekers often struggle to maintain a comprehensive record of their achievements and qualifications, while companies find it challenging to gather and review applicant details efficiently. The reliance on manual processes and physical documents leads to time-consuming and error-prone procedures. This inefficient system may result in missed opportunities and miscommunications between job seekers and companies.

However, our project, the Hiresphere, aims to address these limitations and provide a userfriendly solution. Our platform enables job seekers to easily create profiles and showcase their qualifications and achievements. They can securely store their information, eliminating the need for physical documents and reducing the risk of errors. Similarly, companies can access a centralized database of applicant details, making the hiring process more streamlined and efficient. By leveraging technology, our project revolutionizes the job search and recruitment process, allowing job seekers and companies to connect seamlessly and maximize their opportunities in the digital era.

2.1.1 Limitations of Existing System

- **Limited Job Matches:** Job portals rely heavily on keyword-based search algorithms, which may result in inaccurate or incomplete job matches. This limitation can lead to job seekers missing out on relevant opportunities and companies struggling to find suitable candidates.
- **Lack of Personalization:** Most job portals lack advanced personalization features. They do not consider individual preferences, skills, or career goals when suggesting job opportunities. This can make it difficult for job seekers to find the most suitable positions and for companies to identify the best candidates.
- **Inadequate Company Insights:** Job portals often provide limited information about companies beyond basic details. Job seekers may lack comprehensive insights into the company's culture, work environment, and employee experiences. This can make it

challenging for job seekers to evaluate whether a company is a good fit for their career aspirations.

- **Limited Networking Opportunities:** While job portals offer job listings, they often lack robust networking features. Job seekers may miss out on valuable networking opportunities with industry professionals and potential employers, which can impact their chances of securing desired positions.
- **Difficulty in Verifying Job Listings:** Job portals sometimes struggle to verify the authenticity of job listings, leading to potential scams or misleading opportunities. This poses a risk to job seekers who may unknowingly apply for fraudulent or non-existent jobs.
- **Lack of Real-time Updates:** Job portals may not always provide real-time updates on job availability or application statuses. This can lead to delays in communication between job seekers and companies, causing frustration and missed opportunities.
- **Limited Skill Development Resources:** Job portals primarily focus on job listings and applications, often lacking comprehensive resources for skill development, career guidance, and professional growth. This limits the support available to job seekers in enhancing their skills and advancing their careers.

2.2 PROPOSED SYSTEM

In our project, the Online Job Portal, job seekers and companies benefit from several key features. Job seekers can conveniently access and update their job search history, qualifications, and achievements from any device. The system is designed as a standalone application, ensuring flexibility and security. The efficient design of the application allows for quick retrieval of information, even in urgent situations. The proposed system is highly automated, with simple button clicks facilitating urgent tasks.

Our project focuses on providing a smart appointment booking system for job seekers and companies. Job seekers can easily search for companies based on their specialization and book appointments online. The platform also enables companies to create profiles, allowing job seekers to search and book appointments with them. By offering a web-based application, we overcome the challenges of managing and scheduling appointments according to user preferences.

The existing manual process of scheduling appointments can be tedious and time-consuming for companies. Our project provides an effective solution by allowing users to select a company and book appointments with ease. Through the Online Job Portal, we aim to streamline the job search and hiring process, making it more efficient and convenient for both job seekers and companies.

2.2.1 Advantages of Proposed System:

- **Easy Access and Convenience:** Job seekers can access the Online Job Portal anytime, anywhere, using any device with internet connectivity. This convenience allows for a seamless job search experience, eliminating the need to visit physical job centers or browse through newspapers.
- **Expanded Job Opportunities:** The Online Job Portal provides a vast database of job opportunities from various companies and industries. Job seekers have access to a wide range of positions, increasing their chances of finding suitable employment.
- **Time and Cost Savings:** Job seekers save valuable time and resources by using the Online Job Portal. They can search and apply for jobs efficiently, without the need for extensive paperwork or multiple visits to companies. This streamlined process reduces the overall cost associated with traditional job search methods.
- **Enhanced Job Matching:** The Online Job Portal uses advanced algorithms and search filters to match job seekers with relevant job opportunities. This personalized approach improves the chances of finding a job that aligns with the job seeker's skills, qualifications, and preferences.
- **Company Insights and Research:** The Online Job Portal provides comprehensive information about companies, including their profiles, job descriptions, and requirements. Job seekers can gain insights into company culture, values, and growth opportunities, enabling them to make informed decisions about their career choices.
- **Efficient Application Tracking:** The Online Job Portal allows job seekers to track their applications, ensuring they stay organized and updated throughout the hiring process. This feature eliminates the need for manual follow-ups and provides transparency in application status.
- **Networking Opportunities:** The Online Job Portal often includes networking features, enabling job seekers to connect with industry professionals, mentors, and potential employers. This networking capability expands professional connections and opens doors to new career opportunities.
- **Skill Development Resources:** Many Online Job Portals offer additional resources such as career guidance, resume building tips, interview preparation, and skill development courses. These resources empower job seekers to enhance their skills and improve their chances of securing desired positions.

2.2.2 Features of Proposed System

- Job Search and Filtering: The system allows job seekers to search for relevant job opportunities based on various criteria such as location, industry, job title, and experience level. Advanced filtering options enable job seekers to refine their search and find the most suitable positions.
- Company Profiles: The Online Job Portal provides detailed profiles of companies, including information about their background, culture, mission, and values. Job seekers can gain insights into the company's reputation and make informed decisions about their applications.
- User Profiles: Job seekers can create personalized profiles showcasing their qualifications, skills, work experience, and educational background. This feature enables them to present a comprehensive overview of their professional background to potential employers.
- Job Application Management: The system allows job seekers to manage their job applications efficiently. They can track the status of their applications, receive notifications about application updates, and organize their application history in one centralized platform.
- Resume Builder: The Online Job Portal offers a resume building tool that assists job seekers in creating professional and well-structured resumes. Templates and formatting options are available to ensure the resume highlights the candidate's qualifications effectively.
- Job Alerts and Notifications: Job seekers can subscribe to job alerts and receive notifications about new job openings that match their criteria. This feature helps job seekers stay updated on the latest job opportunities without actively searching the portal.
- Application Tracking: The system provides a dashboard where job seekers can monitor the progress of their applications, including interview schedules, feedback, and final hiring decisions. This tracking feature enhances transparency and keeps job seekers informed throughout the hiring process.

- Communication Channels: The Online Job Portal facilitates communication between job seekers and employers. It may include features such as messaging systems or chatbots to enable direct and convenient communication for clarifying job-related queries or scheduling interviews.

- Skill Assessment and Recommendations: The system may offer skill assessment tests to job seekers, helping them evaluate their strengths and areas for improvement. Based on the results, the system can provide personalized recommendations for skill development resources or relevant job opportunities.

- User Reviews and Ratings: Job seekers can access and contribute to reviews and ratings of companies based on their experiences. This feature allows job seekers to gather insights from other professionals and make informed decisions about potential employers.

2.3 REQUIREMENT SPECIFICATIONS

2.3.1 Hardware Requirements

Processor	:	Intel core i7
Processor speed	:	3GHz or above
RAM	:	3GB or above
Hard Disk Capacity	:	1TB
Keyboard	:	Multimedia Keyboard
Mouse	:	Standard
USB	:	2.0 & 3.0

2.3.2 Software Requirements

Operating System	:	Windows 11
Front End	:	HTML 5, CSS,
Backend	:	Node JS
Language	:	JavaScript
Tools used	:	Visual Studio Code
Database	:	MongoDB

2.3.3 Functional Requirements

A Functional Requirement (FR) is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputs to the software system, its behavior and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. Functional Requirements are also called Functional Specification.

ADMIN MODULE

The admin module controls the overall information and basic credentials of the service system.

FN 1: Admin Login – This function enables the admin to login to the system.

Input: Email and Passwords.
Output: Login successful.

FN 2: View Jobs - The administrator can view the jobs.

Input: Select View jobs option.
Output: Views the list of jobs.

FN 3: Post Jobs - The administrator can post the jobs.

Input: Select Post a job option.
Output: Enters to the Post a Job interface.

FN 4: Delete Jobs- The administrator can delete jobs.

Input: Select Delete option.
Output: Deletes the selected job.

FN 5: View Job Requests- Shows the job application requests from the users.

Input: Select Requests option.
Output: Shows the job requests.

FN 6: Accept Request- Shows the option to accept job application request from the user.

Input: Select Accept option.
Output: Application accepted.

FN 7: Deny Request- Shows the option to deny job application request from the user.

Input: Select Deny option.
Output: Application rejected.

USER MODULE

FN 1: Sign Up– User registers with their details.

Input: Full name, Email, Password, Re-Type password, Mobile Number, Upload Your Resume
Output: Registered successfully.

FN 2: User login – User login with Email and Password.

Input: Email, Password.
Output: Logged in successfully.
FN 3: View Jobs - The user can view the jobs.

Input: Select View jobs option.
Output: Views the list of jobs.
FN 4: Apply for Job- The administrator can apply for the job.

Input: Select apply option.
Output: Applied successfully.
FN 5: Search Job-The administrator can search for specific jobs.

Input: Job type.
Output: Searches the specified Job.

2.3.5 Non-Functional Requirements

In systems, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviour. They are contrasted with functional requirements that define specific behaviour or functions:

- Usability:

Usability defines how difficult it will be for a user to learn and operate the system. Efficiency of use**:** the average time it takes to accomplish a user's goals, how many Tasks a user can complete without any help, the number of transactions completed Without errors.

- Authenticity:

The system meets authenticity as only those users with a valid username and password can log into the application.

- Data integrity:

Data integrity is ensured by only the accurate data.

- Performance:

It ensures good performance by maintaining a low response time, high utilization and fast throughput.

- Availability:

The application meets availability as a user can access it any time.

- Security:

The data is securely stored in database and only the authorized user can access it.

2.4 FEASIBILITY STUDY

Feasibility study is the test of the system proposal made to identify whether the user needs may be satisfied using the current software and hardware technologies, whether the system will be cost effective from a business point of view and whether it can be developed with the given budgetary constraints. A feasibility study should be relatively cheap and done at the earliest possible time. Depending on the study, the decision is made whether to go ahead with a more detailed analysis. When a new project is proposed, it normally goes through feasibility assessment. Feasibility study is carried out to determine whether the proposed system is possible to develop with available resources and what should be the cost consideration. Facts considered in the feasibility analysis were:

- Economic Feasibility
- Operational Feasibility
- Technical Feasibility

2.4.1 Economic feasibility

This feasibility study present tangible and intangible benefits from the prefect by comparing the development and operational cost. This system needs some more initial investment than the existing system, but it can be justifiable that it will improve quality of service.

Thus, feasibility study should centre along the following points:

- Improvement resulting over the existing method in terms of accuracy, timeliness.
- Cost comparison
- Estimate on the life expectancy of the hardware
- Overall objective

Our project is economically feasible. It does not require much cost to be involved in the overall process. It is essential because the main goal of the proposed system is to have economically better result with the increase.

2.4.2 Operational Feasibility

Operational analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and saving that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. An entrepreneur must accurately weigh the cost versus benefits before taking an action. Cost-based study: It is important to identify cost and benefit factors, which can be categorized as follows:

1. Development costs.
2. Operating costs.

This is an analysis of the costs to be incurred in the system and benefits derivable out of the system. Time-based study: This is an analysis of the time required to achieve a return on investments The future value of a project is also a factor. This application can be operated by all the users across the world.

2.4.3 Technical feasibility

Technical Feasibility deals with the hardware as well as software requirements. Technology is not a constraint to type system development. We have to find out whether the necessary technology, the proposed equipment has the capacity to hold the data, which is used in the project, should be checked to carry out this technical feasibility. The technical feasibility issues usually raised during the feasibility stage of investigation includes

- This software is running in Windows 11.
- The minimum hardware required is 3GB and maximum hardware required is 400GB.
- The system can be expanded.

SYSTEM DESIGN

3.1 DESIGN OF SUBSYSTEM

A Design of Subsystem encapsulates behaviour, providing explicit and formal interfaces, and does not expose its internal contents. This provides the ability to completely encapsulate the interactions of a number of classes and/or subsystems. This project has three major modules. These three modules are the core functionalities of the system. The project is divided into three major modules according to their functionalities.

1. ADMIN

2. USERS

1. ADMIN

The main purpose of this module is to view all the registered companies and registered users.

- Admin can view all the registered companies.
- Admin can view all the registered users.
- Admin can view the details about job applications.

2. USERS

The main purpose of this module is that the users can view the details of companies, and make job applications.

- Users can view the details of companies.
- Users can make job applications.

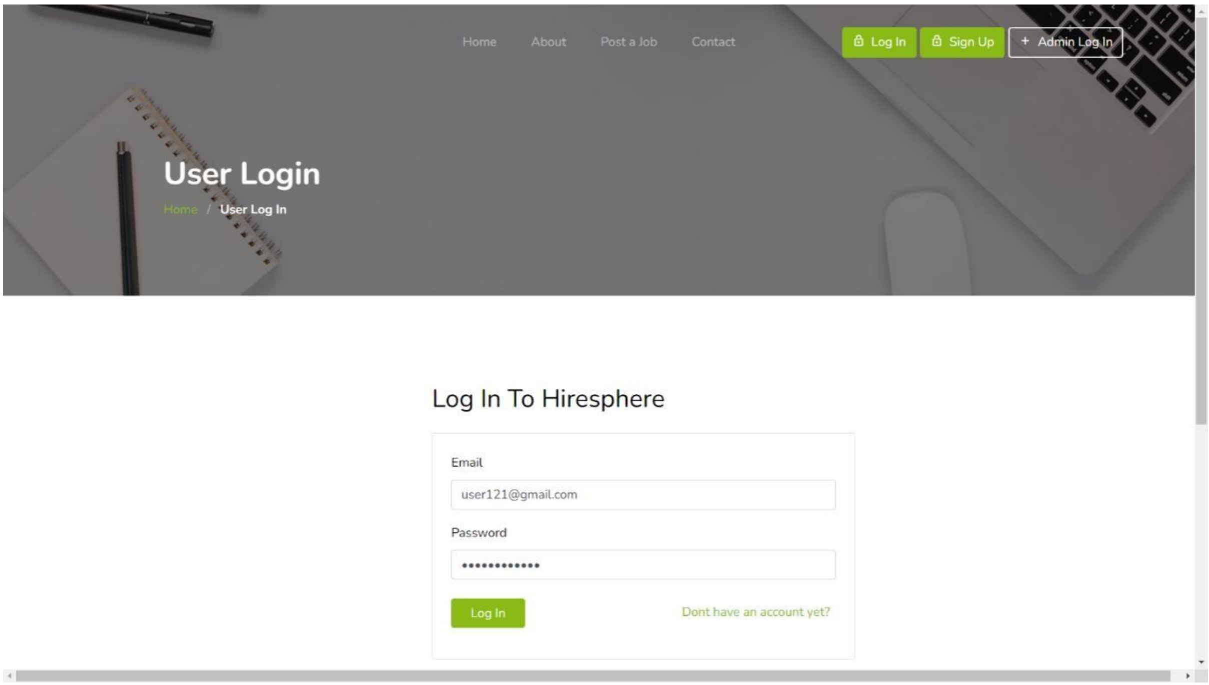
3.2 USER INTERFACE DESIGN

3.2.1 Input Design of User

Input designing is the basic theory to be considered during system study. The input media used in the system is the keyboard. Details are entered in the system through different data entry screens. The system is designed in a user-friendly manner. Appropriate error messages are displayed when a false data is entered. The user interface design is very important for any application. The interface design defines how the software communicates within itself, to system that interpreted with it and with human who use it. The input design is the process of converting the user-oriented description of inputs into a programmer-oriented specification. It is the link that ties the system into the world of its users. Input design involves determining the record media, method of input, speed of capture and entry to the system. The analyst should consider the following points when designing the input:

- Nature of the input processing.
- Flexibility and thoroughness validation rules.
- Handling of priorities with the input procedures.
- Use of composite input documents to reduce the number of different ones.

Login Page



User Login

Log In To Hiresphere

Email

user121@gmail.com

Password

••••••••••••

3.2.2 Output Design of User

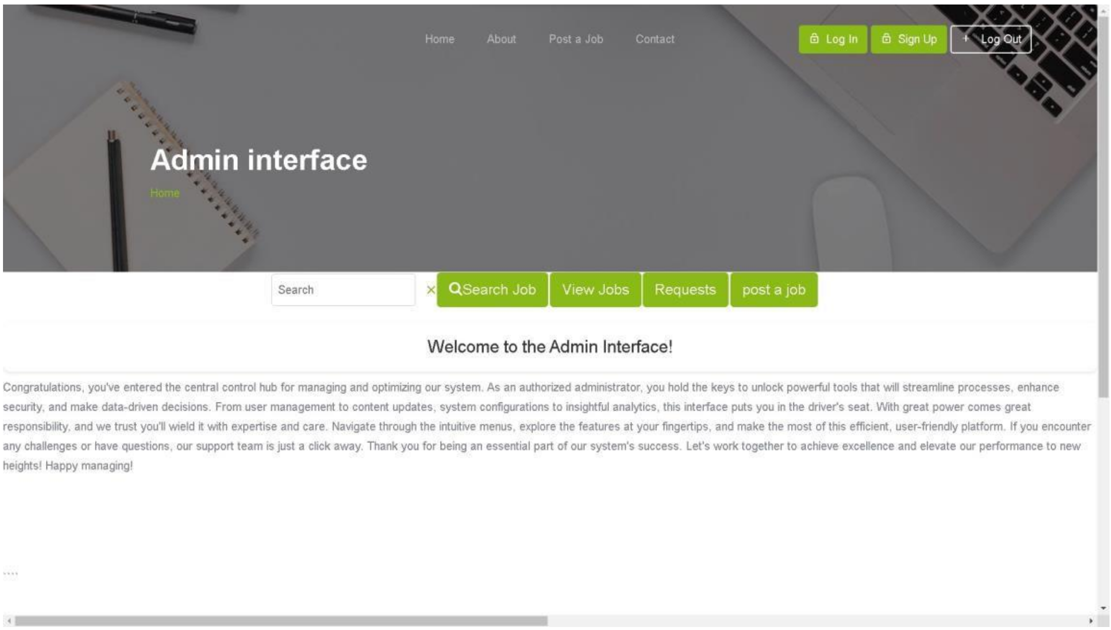
An inevitable activity in the system design is the proper design of output in a form acceptable to the user. Outputs from the system are required primarily to communicate the result of processing to users. Outputs also provide a permanent copy of the results for later consultation. An intelligible output will improve system relationship with the user and help in the decisionmaking process.

The approach to output design is very dependent on the type of output and nature of data. Special attention has to be made to data editing. The choice of appropriate output medium is also an important task. The selection may be affected by the following kinds of consideration:

- Response time.
- Location of users.
- Cost.
- Software/hardware.
- Suitability of the device for application concerned.

The output design must be specified and documented, data items have to be accurately defined and arranged for clarity and easy comprehension.

Admin Interface



Welcome to the Admin Interface!

Congratulations, you've entered the central control hub for managing and optimizing our system. As an authorized administrator, you hold the keys to unlock powerful tools that will streamline processes, enhance security, and make data-driven decisions. From user management to content updates, system configurations to insightful analytics, this interface puts you in the driver's seat. With great power comes great responsibility, and we trust you'll wield it with expertise and care. Navigate through the intuitive menus, explore the features at your fingertips, and make the most of this efficient, user-friendly platform. If you encounter any challenges or have questions, our support team is just a click away. Thank you for being an essential part of our system's success. Let's work together to achieve excellence and elevate our performance to new heights! Happy managing!

3.3 DATABASE DESIGN

The overall objective in the development of database technology has been to treat data as an organizational resource and as an integrated whole. DBMS allow data to be protected and organized separately from other resources. Database is an integrated collection of data. The most significant form of data as seen by the programmers is data as stored on the direct access storage devices. This is the difference between logical and physical data.

Database files are the key source of information into the system. It is the process of designing database files, which are the key source of information to the system. The files should be properly designed and planned for collection, accumulation, editing and retrieving the required information. The organization of data in database aims to achieve three major objectives:

- Data integration.
- Data integrity.
- Data independence.

The proposed system stores the information relevant for processing in the MongoDB database. MongoDB is an open-source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. The NoSQL DBMS uses a single master architecture for data consistency, with secondary databases that maintain copies of the primary database. Operations are automatically replicated to those secondary databases for automatic failover. MongoDB makes use of records which are made up of documents that contain a data structure composed of field and value pairs. Documents are the basic unit of data in MongoDB. The documents are similar to JavaScript Object Notation, but use a variant called Binary JSON (BSON). The benefit of using BSON is that it accommodates more data types. The fields in these documents are similar to the columns in a relational database. Values contained can be a variety of data types, including other documents, arrays and arrays of documents. Documents will also incorporate a primary key as a unique identifier. Sets of documents are called collections, which function as the equivalent of relational database tables. Collections can contain any type of data, but the restriction is the data in a collection cannot be spread across different databases.

Normalization

Normalization is a technique of separating redundant fields and braking up a large table in to a smaller one. It is also used to avoid insertion, deletion and updating anomalies. The table has been normalized up to the third normal form. It is the process of structuring a relational database in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity. Normalization entails organizing the columns (attributes) and tables (relations) of a database to ensure that
their dependencies are properly enforced by database integrity constraints. It is accomplished by applying some formal rules either by a process of synthesis (creating a new database design) or decomposition (improving an existing database design). In short, the rules for each of the five normal forms are as follows: -

- First normal form

  A relation is said to be in 1NF if all the under lying domain of attributes contain simple individual values.

- Second normal form

  The 2NF is based on the concept of full functional dependency. A relation said to be in 2NF if and only if it is in 1NF and every non-key attribute is fully functionally dependent on candidate key of the table.

- Third normal form

  The 3NF is based on the concept of transitive dependency. A relation in 2NF is said to be in 3NF if every non-key attribute is non-transitively dependent on candidate key of the table. Our project is in the second normal form because every non key attribute is fully functionally dependent on primary key. A relation is in third normal form, if there is no transitive dependency for non-prime attributes as well as it is in second normal form.

- Boyce Codd normal form (BCNF)

  BCNF is the advance version of 3NF. It is stricter than 3NF. A table is in BCNF if every functional dependency X → Y, X is the super key of the table. For BCNF, the table should be in 3NF, and for every FD, LHS is super key.

- Fourth normal form (4NF)

A relation will be in 4NF if it is in Boyce Codd normal form and has no multi-valued dependency. For a dependency A → B, if for a single value of A, multiple values of B exist, then the relation will be a multi-valued dependency.

- Fifth normal form (5NF)

A relation is in 5NF if it is in 4NF and not contains any join dependency and joining should be lossless. 5NF is satisfied when all the tables are broken into as many tables as possible in order to avoid redundancy. 5NF is also known as Project-join normal form (PJ/NF).

Our project Hiresphere is based on second normal form (2NF).

3.3.1 Table Design

Admin

Field Name	**Data Type**	**Allow Null**	**Constraints**
Admin email	Varchar(20)	No	Primary key
Admin password	Varchar(20)	No	Not Null

Job Details

Field Name	**Data Type**	**Allow Null**	**Constraints**
Job-id	Int	No	Primary key
Job-type	Varchar(10)	No	Not Null
Job-region	Varchar(10)	No	Not Null
Company email	Varchar(10)	No	Not Null
Company name	Varchar(10)	No	Not Null

User

Field Name	**Data Type**	**Allow Null**	**Constraints**
User-id	Int	No	Primary Key
Full name	Varchar(20)	No	Not Null
User-email	Varchar(20)	No	Not Null
User-password	Varchar(10)	No	Not Null
Mobile number	Int	No	Not Null
Resume	Bindata	No	Not Null

Requests

Field Name	**Data Type**	**Allow Null**	**Constraints**
User-id	Int	No	Primary Key
Full name	Varchar(20)	No	Not Null
Mobile number	Int	No	Not Null
User-email	Varchar(20)	No	Not Null
User-password	Varchar(10)	No	Not Null
Resume	Bindata	No	Not Null

3.3.2 Data Dictionary

Data Dictionary is the major component in the structured analysis model of the system. It lists all the data items appearing in DFD. A data dictionary is a file or a set of files that includes a database's metadata (hold records about other objects in the database), like data ownership, relationships of the data to another object, and some other data**.**

Sl No	Field Name	Description	Data Type	Constraints
1	Admin email	Email of admin	Varchar(20)	Primary key
2	Admin password	Password of admin	Varchar(20)	Not Null
3	Job-id	Id of job	Int	Primary key
4	Job-type	Type of the job	Varchar(20)	Not Null
5	Job-region	Region of the job	Varchar(20)	Not Null
6	Company email	Email of the company	Varchar(20)	Not Null
7	Company name	Name of the company	Varchar(20)	Not Null
8	User-id	Id of the User	Int	Primary Key
9	Full name	Name of the user	Varchar(20)	Not Null
10	User-email	Email of the user	Varchar(20)	Not Null
11	User-password	Password of user	Varchar(20)	Not Null
12	Mobile number	Mobile number of the user	Int	Not Null

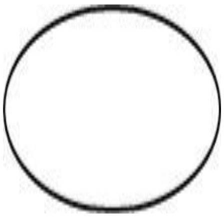
3.4 MODELLING

3.4.1 Data flow diagram (DFD)

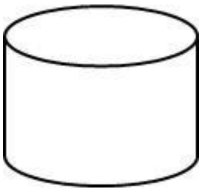
A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design). A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel.

Data Flow Diagram Notations

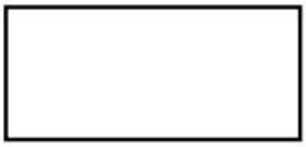
Function:



File/Database:



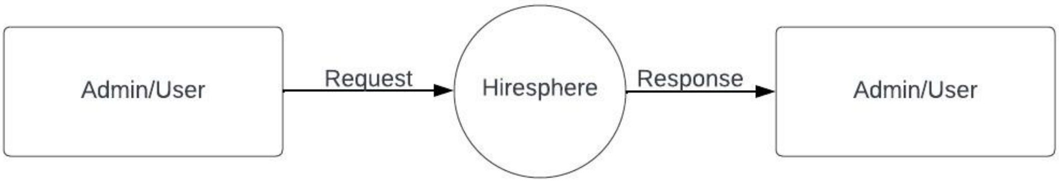
Input/output



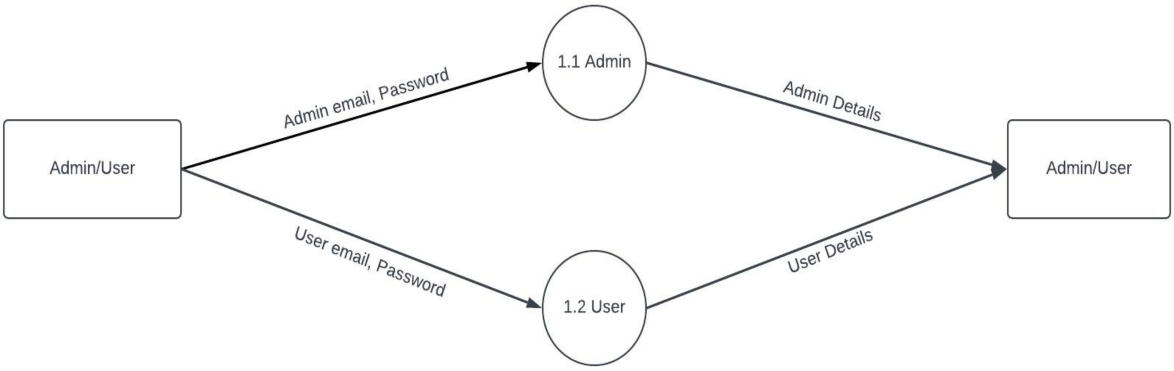
Flow of direction

DATA FLOW DIAGRAM

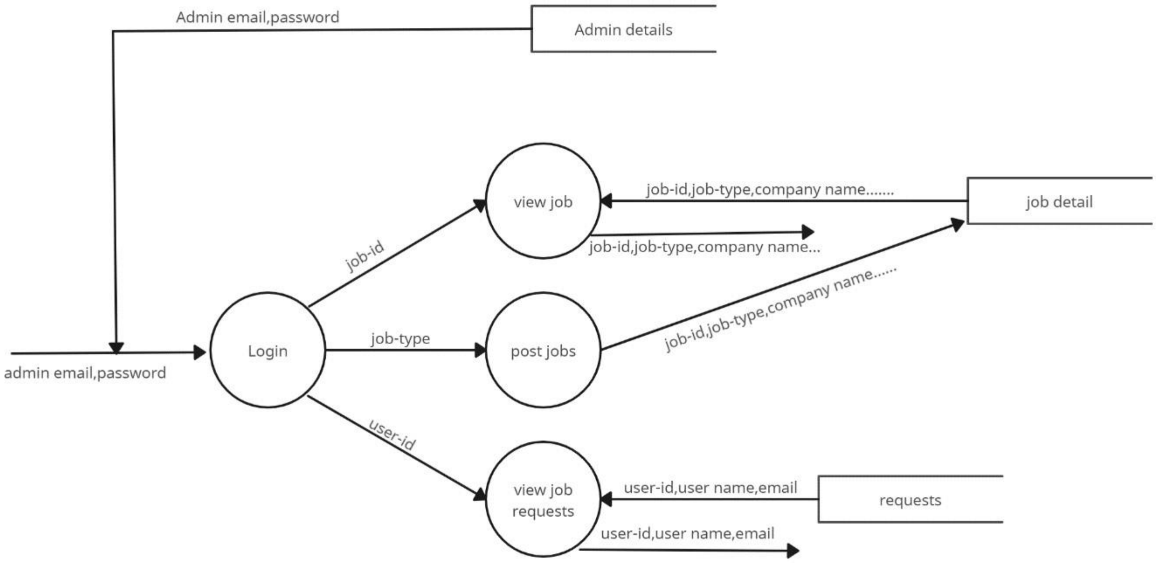
LEVEL-0/CONTEXT LEVEL



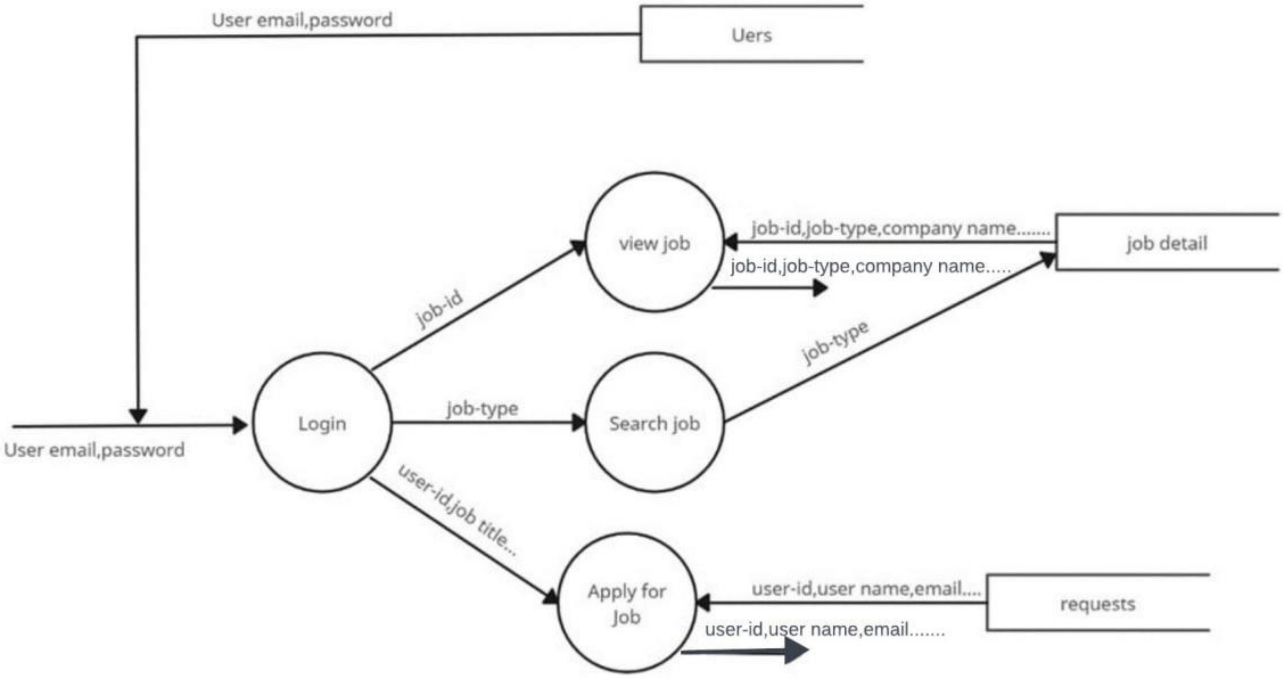
LEVEL-1



LEVEL-1.1 OF ADMIN



LEVEL-1.2 OF USERS



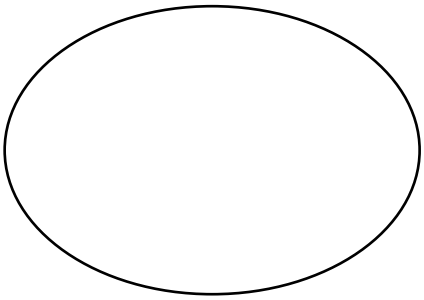
3.4.2 Entity relationship model

An entity relationship diagram is a specialized graphic that illustrates the inter-relationship between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes. An entity– relationship model is the result of using a systematic process to describe and define a subject area of business data. It does not define business process; only visualize business data. The data is represented as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may have various properties (attributes) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity–relationship diagrams.

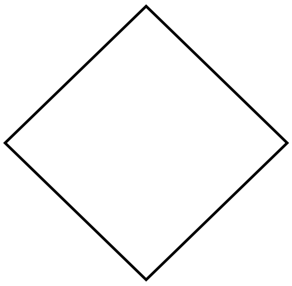
Symbols used in ER diagram: -

Entity:

Attributes:

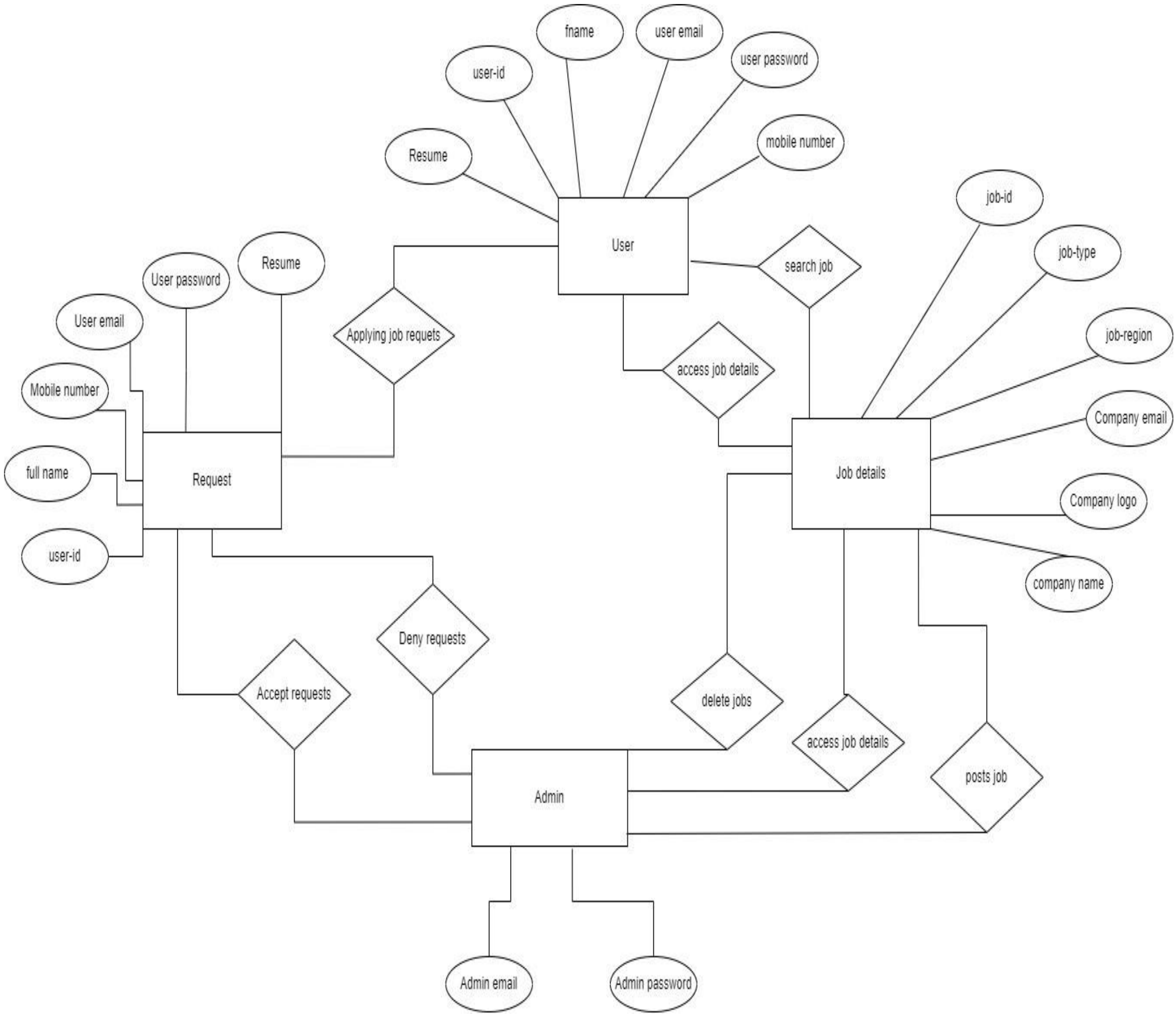


Relationship:



Lines:

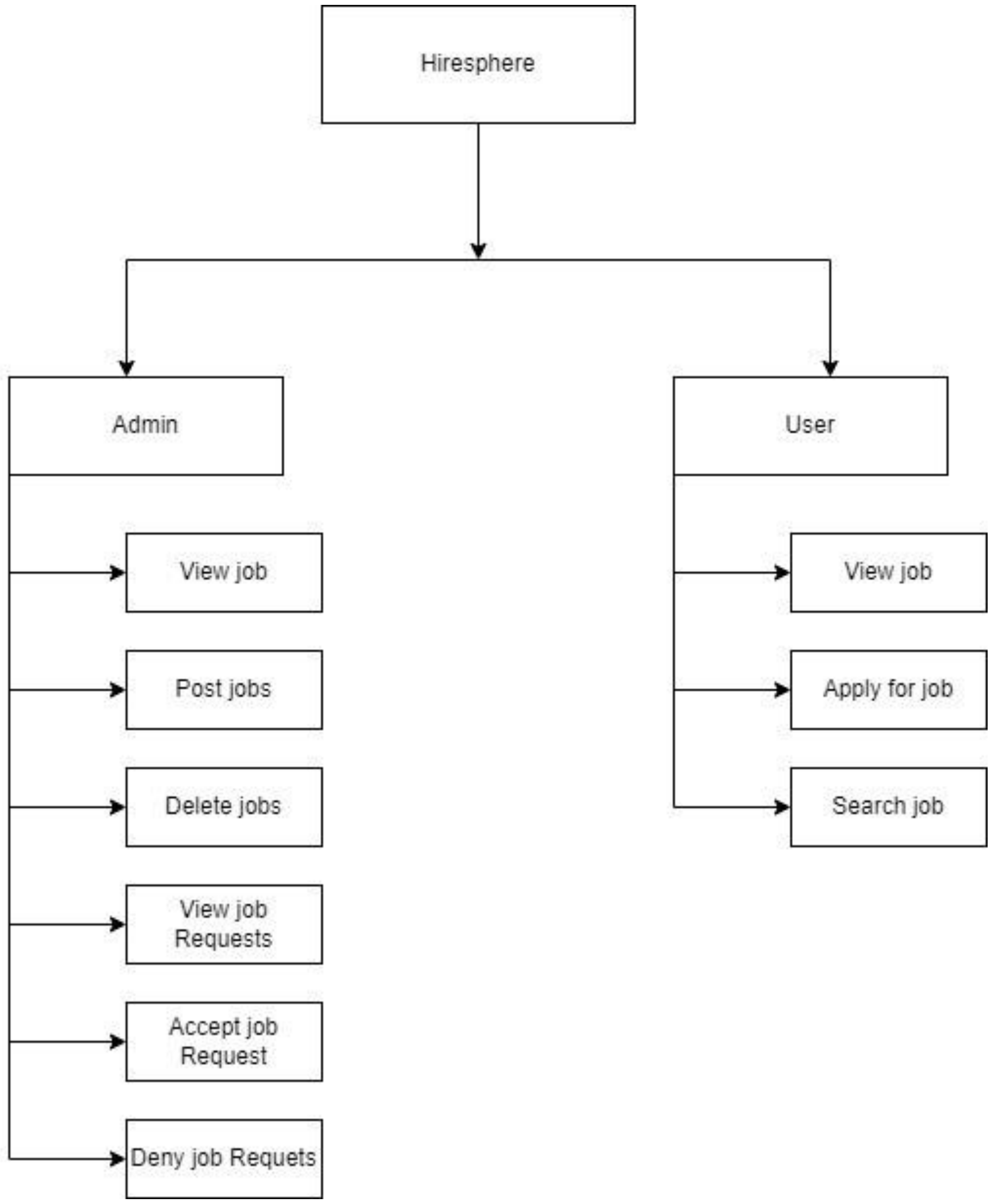
ENTITY RELATIONSHIP DIAGRAM



3.5 ARCHITECTURAL DESIGN

3.5.1 Structure Chart

A structure chart (SC) in software engineering and organizational theory, is a chart which shows the breakdown of a system to its lowest manageable levels. They are used in structure programs to arrange modules into a tree. Each module is represented by a box, which contains the module’s name. The tree structure visualizes the relationships between the modules.



3.6 PROCEDURAL DESIGN

3.6.1 Flow Chart

A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

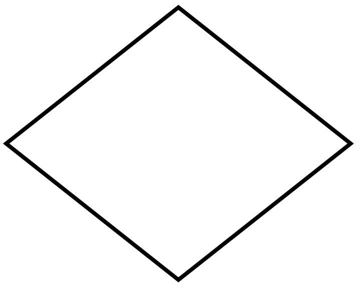
Symbols used:

Flow line:

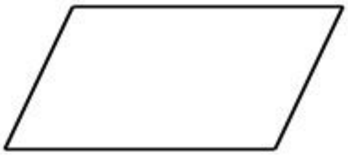
Terminal:

Process:

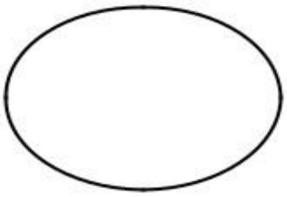
Decision:



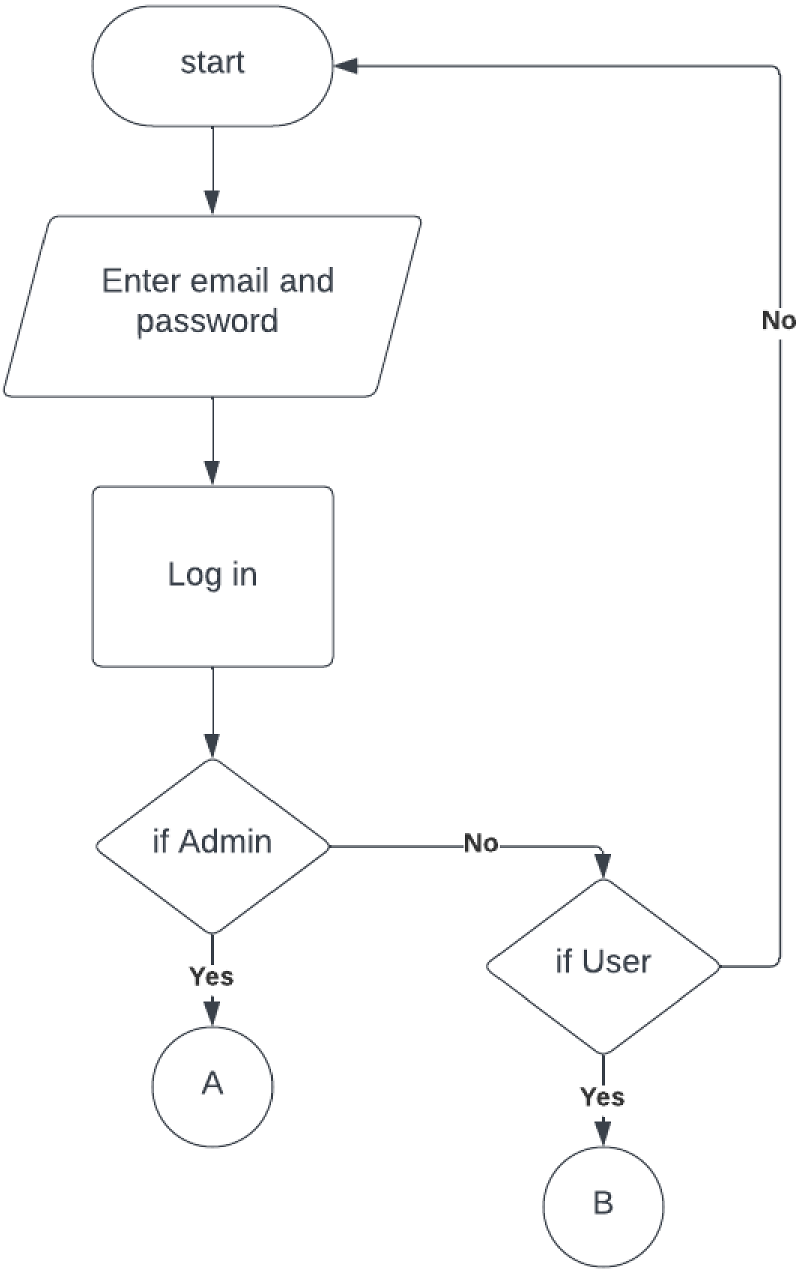
Input/Output:



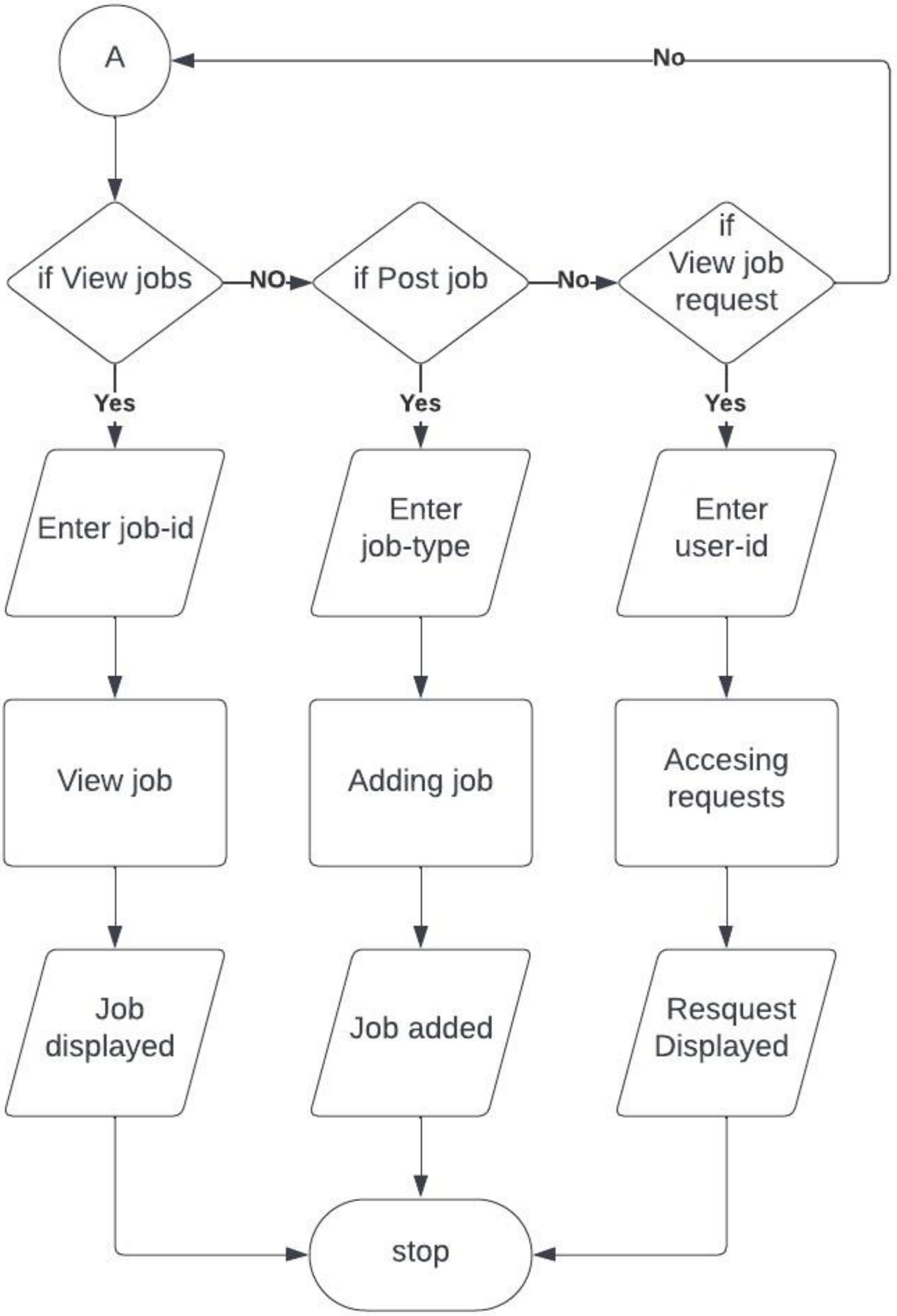
Connector:



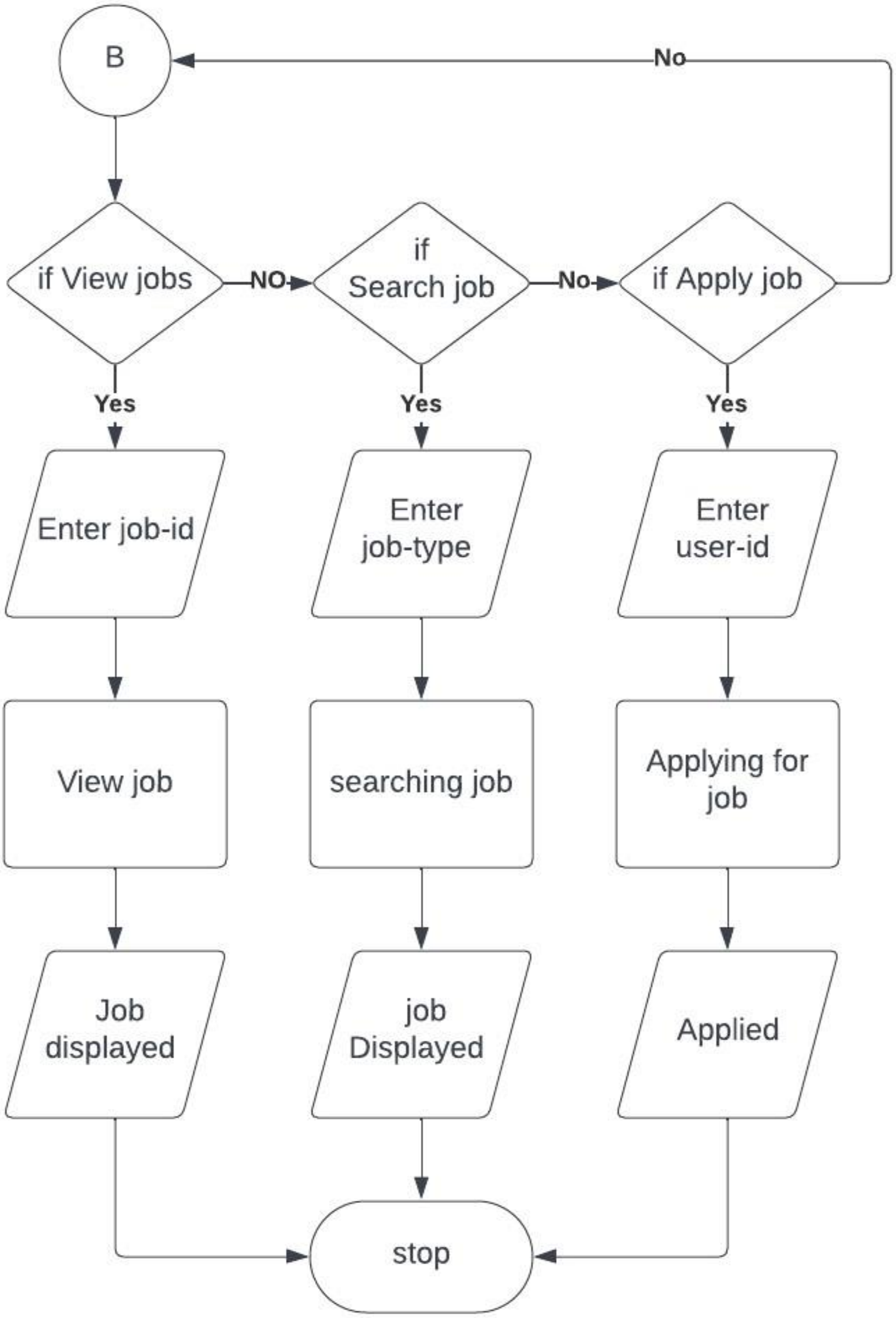
Flow Chart of Hiresphere



Flow chart of Admin



Flow chart of User



PROCESSING ENVIRONMENT

4.1 HARDWARE SPECIFICATIONS

Intel core i5 Processor

An Intel Core i5 is a powerful processor for consumer-end computers and devices. It offers excellent performance and is equipped with Intel Turbo Boost Technology, similar to the Core i7. The Core i5 is available in various versions, ranging from dual-core to hexa-core, allowing for efficient multitasking. It can handle up to 6 threads simultaneously. The processor clock speed of the Core i5 ranges from 1.60 GHz to 4.90 GHz, with cache memory varying from 4 to 12 MB. The thermal design power (TDP) of the Core i5 can range from 15 watts to 125 watts. The Core i5 is well-suited for multitasking, moderate media editing, gaming, and other demanding tasks. It provides a balance between performance and affordability for users.

Processor speed

With technology advancements, increasing productivity goals, faster internet speeds, and the proliferation of devices, the need for speed has become a crucial aspect of our daily lives. We have become accustomed to instant results and expect our devices to keep up with our multitasking demands. When it comes to high-performance technology, computer processors and their clock speed are key factors we consider. The speed of a computer processor, also known as CPU speed, plays a vital role in determining the performance of a computer. The CPU is often referred to as the "brain" of the computer, and its proper functioning is essential for the longevity and functionality of the system. To understand what constitutes a good processor speed, it is important to grasp the role of the processor and how its components enhance the overall functionality of the computer.

RAM

RAM (Random Access Memory) is the hardware in a computing device where the operating system (OS), application programs and data in current use are kept so they can be quickly reached by the device's processor. RAM is the main memory in a computer. This memory stores a temporary data. Temporary data means as long as your computer system is running, your data is stored in RAM and as soon as your computer is shut down, the data stored in RAM gets deleted. RAM is volatile in nature, which means, the data is lost when the device is switched off. RAM is known as the Primary memory of the computer. RAM is known to be expensive since the memory can be accessed directly. RAM is the fastest memory; therefore, it is an internal memory for the computer.

Hard Disk Capacity

A computer hard disk drive (HDD) is a non-volatile data storage device. Non-volatile refers to storage devices that maintain stored data when turned off. All computers need a storage device, and Hard Disk Drive are just one example of a type of storage device. HDD are usually installed inside desktop computers, mobile devices, consumer electronics and enterprise storage arrays in data centres. They can store operating systems, software programs and other files using magnetic disks.

Keyboard

A multimedia keyboard is one with media keys — additional buttons, typically along the top, for controlling audio playback, for starting common applications (e.g., e-mail client and Web browser) and other auxiliary functionality. Many such keyboards also contain a volume knob, implemented as a rotary encoder so as to be able to provide relative volume changes regardless of the volume level set by the user in the operating system and application software.

Mouse

A mouse is a small device that a computer user pushes across a desk surface in order to point to a place on a display screen and to select one or more actions to take from that position. The mouse first became a widely-used computer tool when Apple Computer made it a standard part of the Apple Macintosh.

USB

USB 2.0, also known as hi-speed USB was introduced in 2000. It is an updated version of USB 1.1, which provides improved functionalities and better speed. It is capable to deliver the maximum transfer speed of 480 Megabits per second. However, practically it is approximately 280 Mbps. USB 3.0, also known as SuperSpeed USB was first made available in November 2009. It is a much-improved version of USB 2.0. It supports the data transfer rate of 5 Gigabits per second, which is much faster than the speed provided by USB 2.0.

4.2 SOFTWARE SPECIFICATIONS

Windows 11

Windows 11 is the latest major release of Microsoft's Windows NT operating system, released in October 2021. It is a free upgrade to its predecessor, Windows 10 (2015), available for any Windows 10 devices that meet the new Windows 11 system requirements. Windows 11. A version of the Windows NT operating system.

HTML 5

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and final major HTML version that is a World Wide Web Consortium recommendation. The current specification is known as the HTML Living Standard.

CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Node JS

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine and executes JavaScript code outside a web browser, which was designed to build scalable network applications. Node.js is primarily used for non-blocking, event-driven servers, due to its single-threaded nature. It's used for traditional web sites and back-end API services, but was designed with real-time, push-based architectures in mind.

JavaScript

JavaScript, often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use JavaScript on the client side for web page behaviour, often incorporating third-party libraries. JavaScript runs on the client side of the web, which can be used to design / program how the web pages behave on the occurrence of an event. JavaScript is an easy to learn and also powerful scripting language, widely used for controlling web page behaviour. It is used to enhance HTML pages and is commonly found embedded in HTML code. JavaScript is an interpreted language. Thus, it doesn't need to be compiled. JavaScript renders web pages in an interactive and dynamic fashion.

Visual Studio Code

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, Mac-OS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.

MongoDB

MongoDB is an open source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, store or retrieve information. MongoDB supports various forms of data. It is one of the many nonrelational database technologies that arose in the mid-2000s under the NoSQL banner -- normally, for use in big data applications and other processing jobs involving data that doesn't fit well in a rigid relational model. Instead of using tables and rows as in relational databases, the MongoDB architecture is made up of collections and documents.

CODING AND IMPLEMENTATION

5.1 CODING

AdminLogin (Admin-login.html)

```
<div class="container">

    <div class="row">

      <div class="col-md-7">

        <h1 class="text-white font-weight-bold">Admin login</h1>

        <div class="custom-breadcrumbs">

          <a href="#">Home</a> <span class="mx-2 slash">/</span>

          <span class="text-white"><strong>Admin Log In</strong></span>

        </div>

      </div>

    </div>

  </div>

 </section>


<!-- ...existing code... -->


<section class="site-section">

  <div class="container">

   <div class="row">

     <div class="col-lg-6" style="margin-left: 30%;">

      <h2 class="mb-4">Log In To Hiresphere</h2>

      <form action="#" class="p-4 border rounded" onsubmit="login(event)">
```

```
        <div class="row form-group">
          <div class="col-md-12 mb-3 mb-md-0">
            <label class="text-black" for="email">Email</label>
            <input type="text" id="email" class="form-control" placeholder="Email address" required>
          </div>
        </div>
        <div class="row form-group mb-4">
          <div class="col-md-12 mb-3 mb-md-0">
            <label class="text-black" for="password">Password</label>
            <input type="password" id="password" class="form-control" placeholder="Password" required>
          </div>
        </div>

        <div class="row form-group">
          <div class="col-md-12">
            <input type="submit" value="Log In" onclick=" login(event)" class="btn px-4 btn-primary text-white">
          </div>
        </div>

      </form>
    </div>
```

```
      </div>

     </div>
```

User Registration(signup.html)

```
<div class="container">

    <div class="row">


     <!-- signup -->

     <div style="margin-left:30% ;" class="col-lg-6 mb-5">

      <h2 class="mb-4">Sign Up To Hiresphere</h2>

      <form action="#" class="p-4 border rounded">

        <div class="row form-group">

          <div class="col-md-12 mb-3 mb-md-0">

           <label class="text-black" for="fname">Full Name</label>

           <input type="text" id="fname" class="form-control" placeholder="Full Name">

          </div>

         </div>

       <div class="row form-group">

        <div class="col-md-12 mb-3 mb-md-0">

         <label class="text-black" for="email">Email</label>

         <input type="email" id="email" class="form-control" placeholder="Email address">

        </div>

       </div>

       <div class="row form-group">
    <div class="col-md-12 mb-3 mb-md-0">

      <label class="text-black" for="password id">Password</label>

      <input type="password" id="password" class="form-control" placeholder="Password">

    </div>

  </div>

  <div class="row form-group mb-4">

    <div class="col-md-12 mb-3 mb-md-0">

      <label class="text-black" for="password re id">Re-Type Password</label>

      <input type="password" id="passwordr" class="form-control" placeholder="Re-type Password">

    </div>

  </div>

  <div class="row form-group">

    <div class="col-md-12 mb-3 mb-md-0">

      <label class="text-black" for="mid">Mobile Number</label>

      <input type=" number" minlength="10" id="mid" class="form-control" placeholder="Mobile
Number">

    </div>

  </div>

  <div class="form-group">

    <label for="company-website-tw d-block"> Upload Your Resume</label> <br>

    <label class="btn px-4 btn-primary text-white" id="signupbtn">Upload<input type="file"  hidden>

    </label>
```

```
      <div class="row form-group">
        <div style="margin-left: 35%;" class="col-md-12">
          <button type="submit" onclick="Register()" class="registerbtn">Register</button>
        </div>
      </div>
    </div>
    <div class="row form-group" style="margin-left: 27%;">
        <div class="col-md-12">
          <a  href="./login.html">Already have an account?</a>
        </div>
      </div>
    </form>
  </div>
</div>
</div>
```

User(user.js)

```
 var mongoose=require('mongoose'); var
UserSchema=new mongoose.Schema(
    {
fname:String,
email:String,
password:String,
passwordr:String,
mid:String
    },
    {timestamps:true}
);
module.exports=mongoose.model('User',UserSchema,'User');
```

Postjob(site.js)

```
 var mongoose=require('mongoose'); var SiteSchema=new
mongoose.Schema(
    {
email:String,
jobregion:String,
jobtype:String,
jdesc:String,
comname:String,
comtag:String,
comdesc:String,
comweb:String,
link:String
    },
    {timestamps:true}
);
module.exports=mongoose.model('Site',SiteSchema,'Site');
```

Index.js

```
 const express=require('express');//for accessing the package express const
mongoose=require('mongoose');//for accessing the package mongoose const path =
require('path'); const user = require('./user.js'); const User=require("./user.js"); const site =
require('./site.js'); const Site = require("./site.js");
mongoose.connect('mongodb+srv://sidworld312:sivankutty69@cluster0.3f2xe53.mongodb.n
et/signup'); const app=express();//to call express
package app.use(express.json());//use the package
express let PORT = 3000//assigning the port numb
app.use(express.static(path.join(__dirname, 'public')));
app.use(function (req, res, next) {


  // Website you wish to allow to connect
  //res.setHeader('Access-Control-Allow-Origin','http://localhost:4200');
res.setHeader('Access-Control-Allow-Origin', 'http://localhost:5000');


  // Request methods you wish to allow   res.setHeader('Access-Control-Allow-Methods',
'GET, POST, OPTIONS, PUT, PATCH, DELETE');


  // Request headers you wish to allow
  // res.setHeader('Access-Control-Allow-Headers', 'X-Requested-With,content-type');
  res.header("Access-Control-Allow-Headers", "Origin, X-Requested-With, Content-Type,
Accept, Authorization");
```

```
  // Set to true if you need the website to include cookies in the requests sent
  // to the API (e.g. in case you use sessions)    res.setHeader('Access-Control-Allow-
Credentials', true);

  // Pass to next layer of middleware   next(); });
app.post('/login',(req,res)=>{console.log(req.body)
console.log(req.query)   login(req,res)
})
app.post('/signup',(req,res)=>{console.log(req.body)
console.log(req.body)
res.send({"status":200,"data":"User created Sus"})
User.create(req.body);

})
app.post('/postajob',(req,res)=>{console.log(req.body)
console.log(req.body)
res.send({"status":200,"data":"Site created Sus"})
  Site.create(req.body);
}) app.get('/view', (req, res) =>
{console.log(req.body)   console.log(req.query)
fetchSite(req,res)
})
```

```
//admin app.get('/viewadmin', (req, res) =>

{console.log(req.body)   console.log(req.query)

fetchSite(req,res)

})

//user app.get('/viewuser', (req, res) =>

{console.log(req.body)   console.log(req.query)

fetchSite(req,res)

}) app.get('/', (req, res)

=> {   res.send("ok");

  })

  async function login(req,res){     var

data=await user.findOne(req.body);

data=data||0;     if(data!=0){

res.send({"status":200,"data":"valid user"})

    }     else(

res.send({"status":404,"data":"invalid user"})

    )

  }

  //Deleteuser

app.delete('/deleteuser',(req,res)=>{

console.log(req.query);

console.log(req.body);     datadelete(req,res)
}) async function datadelete(req, res) {   try {
const email = req.query.email || req.body.email;
await Site.deleteOne({ id: email });     res.json({
message: 'User deleted successfully' });
  } catch (error) {     console.error(error);
res.status(500).json({ message: 'Failed to delete the User' }); }
} app.get('/sentreq',async(req,res)=>{   console.log(req.body)
console.log(req.query)   fetchrentreq(req, res)
}) async function fetchrentreq(req, res) {
var data = await Memb.find();
res.send({ "status": 200, "data": data });
} app.put('/',(req,res)=>{
console.log(req.query);
console.log(req.body);
dataupdate(req,res)
})   async function dataupdate(req,res){     console.log(req.query);
console.log(req.body);     res.send(await
User.updateOne({"id":req.query.id},{$set:req.body}))
}
  //   async function datadelete(req,res){
  //     console.log(req.query);
  //     console.log(req.body);
  //     res.send(await User.deleteOne({"id":req.query.id}))
  // }
  async function fetchSite(req,res){
var data = await Site.find();
res.send({"status":200,"data":data});
  }
  //admin async function
fetchSite(req,res){   var data = await
Site.find();
res.send({"status":200,"data":data});
}     app.delete('/user',(reg,res)=>{res.send("DELETE     LIST");});
app.listen(PORT,() =>console.log(`Listening on port${PORT}..`));
```

5.2. TESTING

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also. The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works, but the intent should be to show that a program doesn't work. Testing is the process of executing a program with the intent of finding errors.

5.2.1 Unit Testing

Unit testing plays a crucial role in our project, focusing on verifying the functionality of individual components or modules. Each service within the online job portal can be considered as a separate module, including the Admin, Job Seeker, and Company modules. During the development phase, unit testing is performed to ensure the proper functioning of each module without any errors.The inputs received from users are thoroughly validated to ensure their accuracy and adherence to the system requirements. As part of unit testing, the developer tests each program within the system. Software units in the online job portal are represented by modules and routines that are integrated to perform specific functions.By conducting unit testing on individual modules, we can detect and address errors before they propagate to other parts of the system. This approach allows us to identify and resolve any issues resulting from interactions between different modules, ensuring a smooth and error-free functioning of the online job portal.

Black Box Testing: In the black box testing, test cases are designed from the examination of the input/output values only and no knowledge of design or code is required. The main approaches are:

- Equivalent class partitioning
- Boundary value analysis

White Box Testing: White box testing is an important type of unit testing. A large number of white box testing strategies exist. White box testing strategies are:

- Fault based testing
- Coverage Based testing

5.2.2 Integration Testing

Integration testing is a type of software testing in which the different units, modules or components of a software application are tested as a combined entity. However, these modules may be coded by different programmers. The aim of integration testing is to test the interfaces between the modules and expose any defects that may arise when these components are integrated and need to interact with each other.

Types of test cases

Integration test case: Here we execute test cases which just tell about the connectivity from one module to other module and integrating one application to other application and how the application moves from parent node to child no demand vice versa.

Functional Test case: Here we execute test cases which tell about the functionality of the application and talks about the desired output to be seen. Internally we have different type of test cases when we write here like range, output values, BVA, ECP and so on. We give input and expect some output according to the SRS. Here we check individual module completely by checking each and every tab, text box, and buttons and so on.

Non-Functional Test case: Test cases related to user friendliness like font, image, color, how easy to navigate etc., performance related, security related comes under here.

User Acceptance test case: These test cases are crucial and very important to client-side people, because these test case talks about these business and approach of the application to complete a particular client task, which is also called as End to End Business scenario test case.

Here we won’t be doing testing related to UI, Functional or Non-Functional, we talk about business and scenario for which the application is made for.

5.2.3 System Testing

Here the entire software system is tested. The reference document for this process is the requirements document, and the goal is to see if software meets its requirements. The entire software has been tested against requirements of project and it is checked whether all requirements of project have been satisfied or not.

Alpha Test: The first test of newly developed, when the first round of bugs has been fixed, the product goes with actual users for testing. For custom software the customer may be invited into the vendor's facilities for an alpha test to ensure the client's vision has been interpreted properly by the developer.

Beta Test: A test of new or revised software application that is performed by users at their facilities under normal operating conditions. Beta testing follows alpha testing. Vendors of packaged software often offer their customers the opportunity of beta testing new releases or versions and the beta testing of elaborate products such as operating systems can take months.

Acceptance testing: Testing is performed by the Client of the application to determine whether the application is developed as per the requirements specified by him/her. It is performed within the development of the organization or at the client site.

In our project the developer uses unit testing to ensure the overall working of the system.

SECURITY, BACKUP AND RECOVERY MECHANISM

Security

This application deals with a simple authorization mechanism i.e., username and password. The user just has to register into the application with his username and password. The application can only be operated by a registered user and with his account only. Admin can accept or decline the user request based on the authenticity of the data entered.

Backup

In any case where the user encounters a problem with the application, support will be provided to the user for fixing the problem. A backup source code for our application is always kept.

Recovery

In any case the application crashes the application just need to be refreshed to be back online. In worst case refreshing the system might prove helpful and solves the problem. If the application has any problems, there are other recovery options as well.

FUTURE ENHANCEMENT

In the future, our online job portal system can be expanded to encompass multiple companies and job seekers, extending its reach beyond a specific organization. This would allow for intercompany data sharing, making it easier to refer job seekers to other companies when suitable opportunities arise. By hosting the platform on online servers, we can ensure global accessibility, enabling users from anywhere to utilize the system.Currently, the system facilitates job applications and listings. In the future, we can enhance the platform by integrating video conferencing capabilities, such as Google Meet, enabling job seekers to have virtual interviews and consultations with potential employers from the comfort of their homes. To improve communication, we can implement SMS alerts for upcoming interviews and application statuses. Moreover, we can incorporate online payment options, including credit/debit cards, UPI, and internet banking, allowing job seekers to make payments directly for services like resume writing or premium job listings. Similar to e-commerce platforms like Amazon, we can introduce a provision for job seekers to purchase relevant resources or courses to enhance their skills. To facilitate informed decision-making, we can introduce a rating system where job seekers can rate their experience with employers, providing valuable insights for other job seekers. Additionally, implementing time slots for interviews or interactions can prevent overcrowding during specific periods, ensuring a smoother process for both companies and job seekers. Furthermore, we can indicate the online/offline status of companies and job seekers and provide a chat box feature, allowing real-time communication for inquiries and clarifications when the respective parties are available. These enhancements will create a more comprehensive and user-friendly online job portal experience.

SOFTWARE MAINTENANCE

Software maintenance is widely accepted part of SDLC now days. It stands for all the modifications and update done after the delivery of software product.

Software Maintenance planning includes ten activities:

- **Preparation** – Describe software preparation and transition activities including the conception and creation of the maintenance plan; describe how to handle problems identified during development and configuration management.
- **Modification** – After the application has become the responsibility of the maintenance team, explain how to analyse each request; confirm and check validity; investigate and propose solutions; document the proposal and get the required authorizations to apply the modifications.
- **Implementation** – Describe the process for considering the implementation of the modification itself.
- **Acceptance** – Describe how the modification is accepted by the maintenance team.
- **Migration** – Describe any migration tasks that need to be executed. If the software needs to be moved to another system, outline the steps to do so without impacting its functionality.
- **Transition** – Document the sequence of activities to transition the system from Development to Maintenance.
- **Service Level Agreements** – Document SLAs and maintenance contracts negotiated by Maintenance.
- **Change Request** – Outline the problem-handling process to prioritize, documents and route change and maintenance requests.
- **Modification Request acceptance**/**reject**– Describe the request including details of the size/effort/complexity. If this is too complex to resolve, outline the steps to route the issue back to the software team.
- **Retirement** – This is the final stage in the lifecycle. Describe how to retire the software and the steps to archive any data that may be a by-product of the system.

Types of Maintenance

Traditionally, 5 types of maintenance have been distinguished, which are differentiated by the nature of the tasks that they include:

Corrective maintenance: The set of tasks is destined to correct the defects to be found in the different equipment and that are communicated to the maintenance department by users of the same equipment.

Preventive Maintenance: Its mission is to maintain a level of certain service on equipment, programming the interventions of their vulnerabilities in the most opportune time. It is used to be a systematic character, that is, the equipment is inspected even if it has not given any symptoms of having a problem.

Predictive Maintenance: It pursues constantly know and report the status and operational capacity of the installations by knowing the values of certain variables, which represent such state and operational ability. To apply this maintenance, it is necessary to identify physical variables (temperature, vibration, power consumption, etc.) and to understand which type of variation is indicative of problems that may be appearing on the equipment. This maintenance it is the most technical, since it requires advanced technical resources, and at times of strong mathematical, physical and / or technical knowledge.

Zero Hours Maintenance (Overhaul): The set of tasks whose goal is to review the equipment at scheduled intervals before appearing any failure, either when the reliability of the equipment has decreased considerably so it is risky to make forecasts of production capacity. This review is based on leaving the equipment to zero hours of operation, that is, as if the equipment were new. These reviews will replace or repair all items subject to wear. The aim is to ensure, with high probability, a good working time fixed in advance.

Periodic maintenance (Time Based Maintenance TBM):

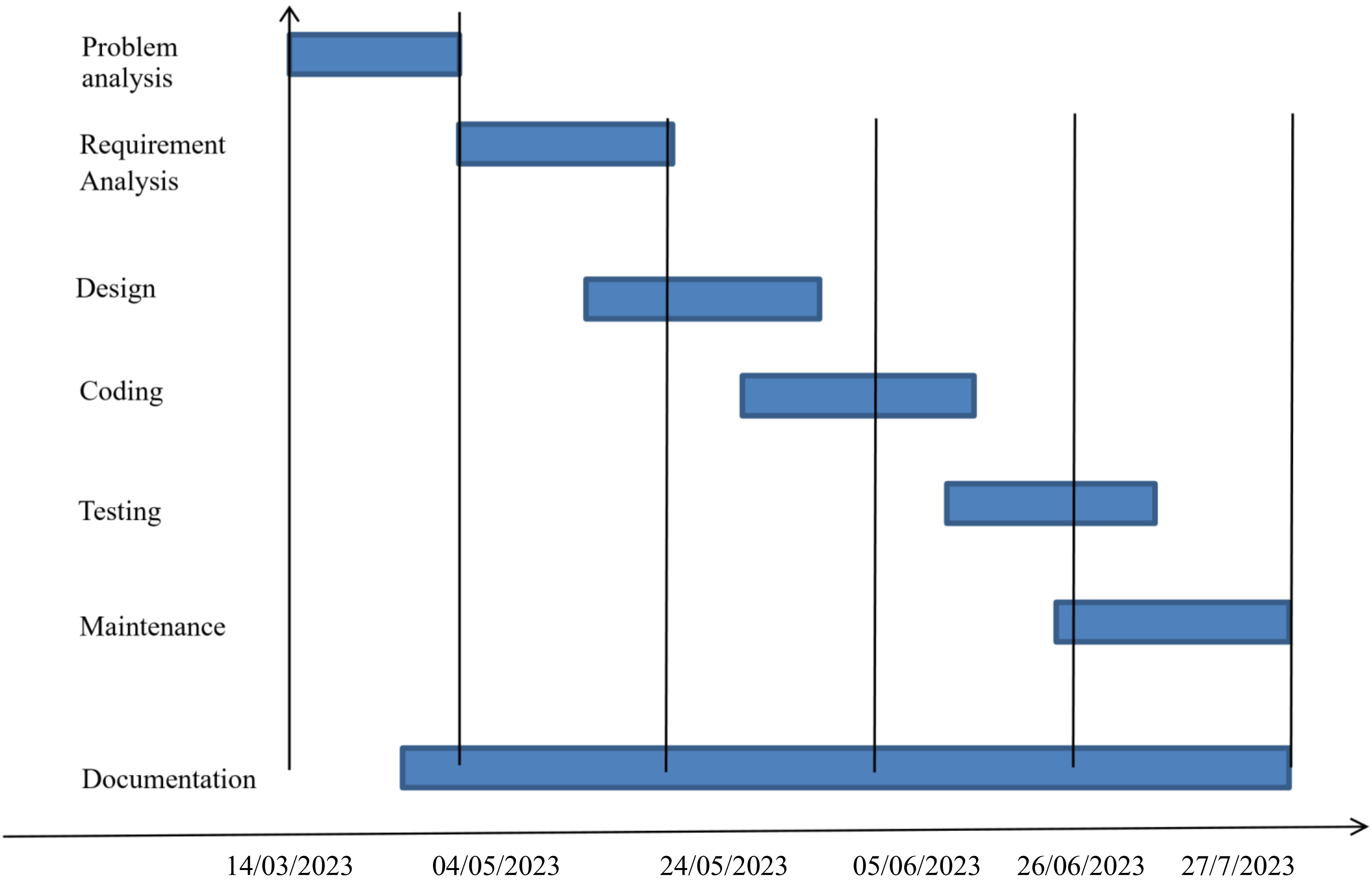
The basic maintenance of equipment made by the users of it. It consists of a series of elementary tasks (data collections, visual inspections, cleaning, and lubrication, retightening screws) for which no extensive training is necessary, but perhaps on brief training. This type of maintenance is the based on TPM (Total Productive Maintenance).

CONCLUSION

We propose the development of an innovative online job portal application to streamline the job search and recruitment process. Our system aims to provide a user-friendly interface for both job seekers and companies, facilitating efficient job matching and hiring. This standalone application can be accessed by job seekers and companies through their desktop or mobile devices.Upon installation, job seekers can create their profiles, upload their resumes, and search for job opportunities based on their skills and preferences. The application allows job seekers to explore various job listings, view company profiles, and apply for positions seamlessly. Companies, on the other hand, can create profiles, post job openings, and review applicant profiles to identify potential candidates.Our system ensures the secure storage and management of job seeker and company data. It allows for easy search and filtering options, enabling job seekers to find relevant job openings and companies to identify suitable candidates. The application can be used in public places with Wi-Fi connectivity, and efforts are underway to make it available offline as well.To increase awareness and user adoption, we plan to promote the application through meetups, awareness programs, and marketing campaigns. Additionally, we aim to make the application accessible through popular app stores like Google Play and Apple Store.While the initial version of our application offers a range of essential features, we have plans to incorporate additional functionalities based on user feedback and requirements. These may include video conferencing for virtual interviews, an online marketplace for job-related resources, and a rating system to evaluate companies based on their hiring processes.With its reliability, efficiency, and potential for future enhancements, our online job portal application aims to revolutionize the job search and recruitment landscape, connecting job seekers and companies in a seamless and convenient manner.

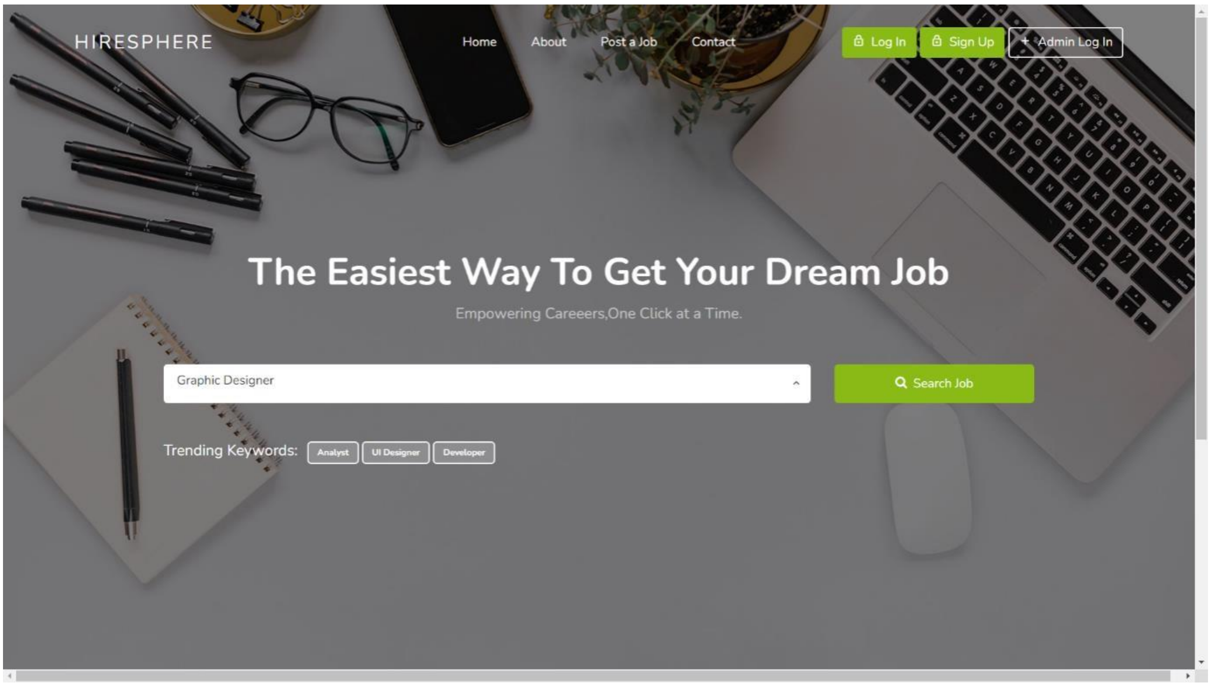
APPENDIX

GANTT CHART


Problem analysis
Requirement Analysis
Design
Coding
Testing
Maintenance
Documentation
14/03/2023
04/05/2023
24/05/2023
05/06/2023
26/06/2023
27/7/2023

SCREENSHOTS

Homepage

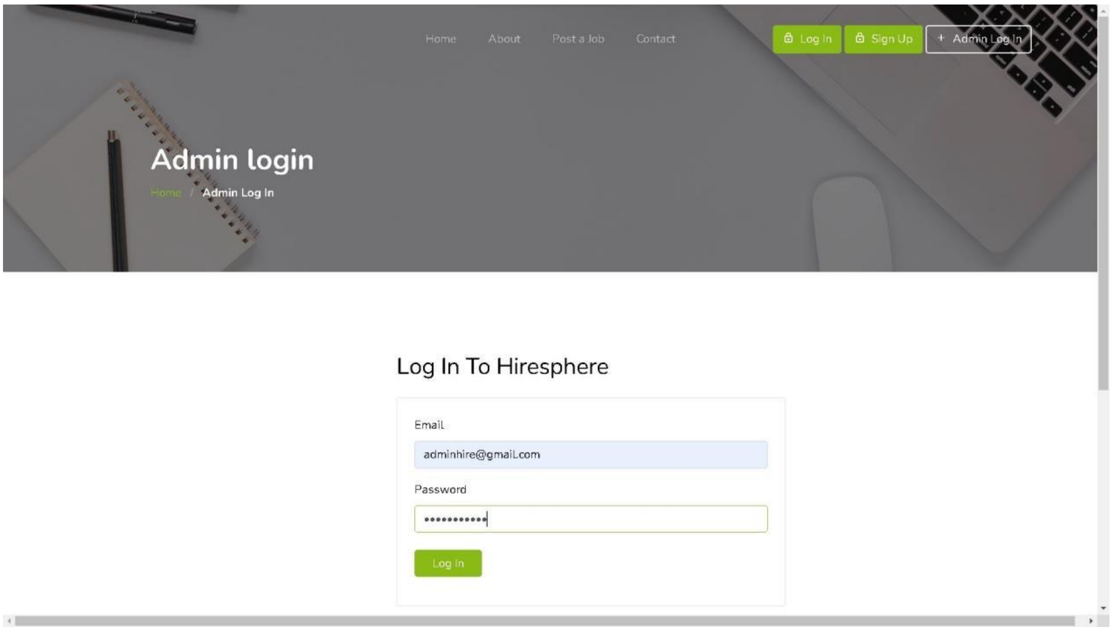


The Easiest Way To Get Your Dream Job

Empowering Careeers,One Click at a Time.

Graphic Designer

Admin Login Page



Admin login

Log In To Hiresphere

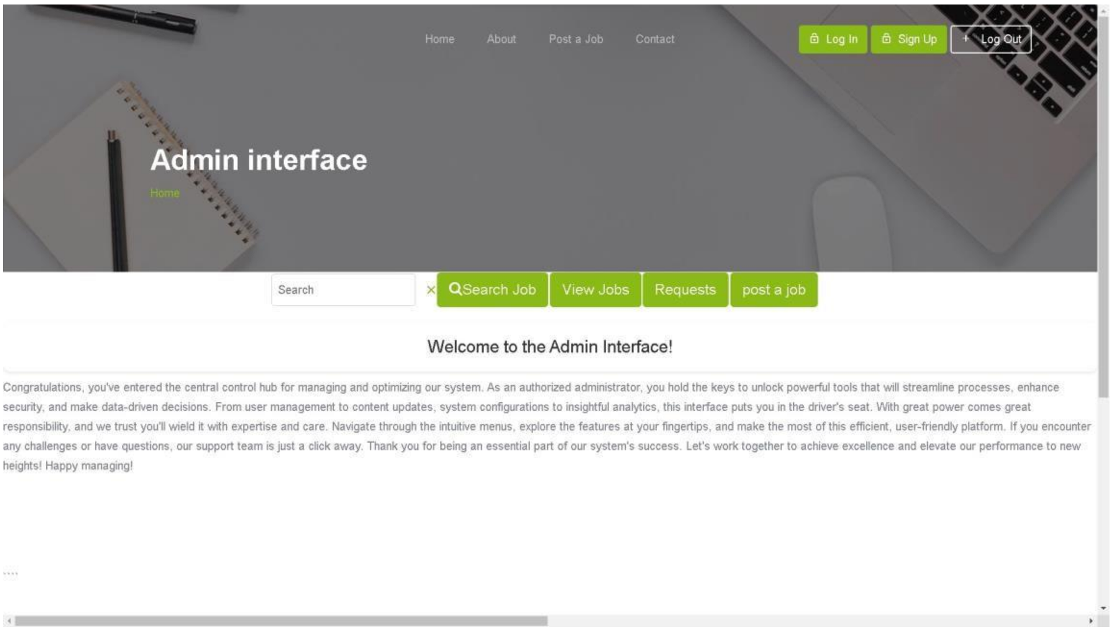
Email

adminhire@gmail.com

Password

•••••••••••

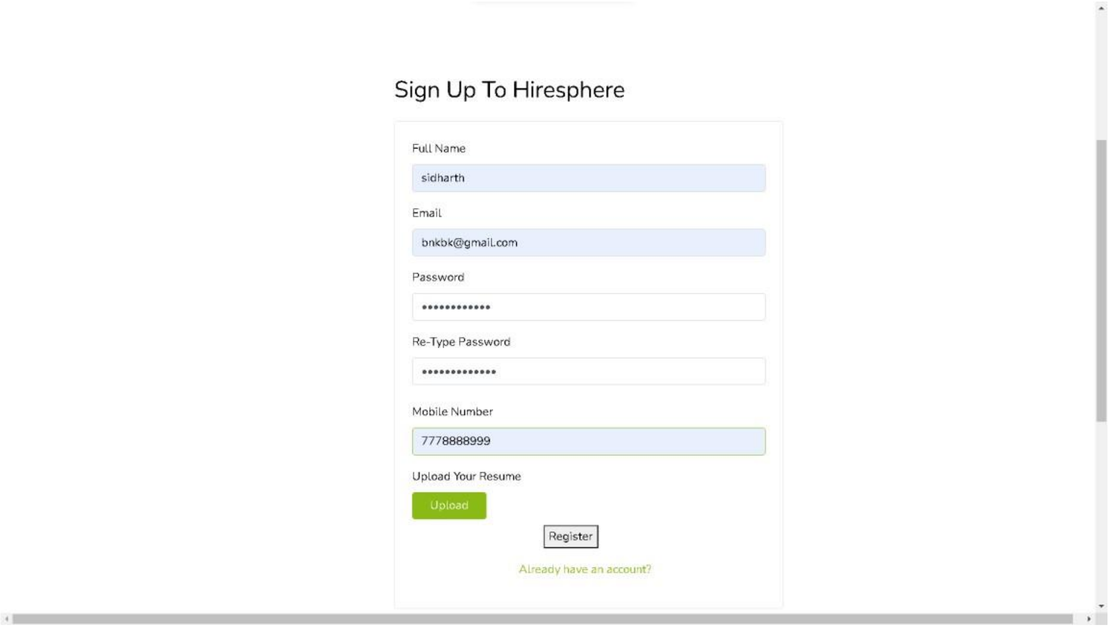
Admin Interface



Welcome to the Admin Interface!

Congratulations, you've entered the central control hub for managing and optimizing our system. As an authorized administrator, you hold the keys to unlock powerful tools that will streamline processes, enhance security, and make data-driven decisions. From user management to content updates, system configurations to insightful analytics, this interface puts you in the driver's seat. With great power comes great responsibility, and we trust you'll wield it with expertise and care. Navigate through the intuitive menus, explore the features at your fingertips, and make the most of this efficient, user-friendly platform. If you encounter any challenges or have questions, our support team is just a click away. Thank you for being an essential part of our system's success. Let's work together to achieve excellence and elevate our performance to new heights! Happy managing!

User Registration Page



Sign Up To Hiresphere

Full Name: sidharth

Email: bnkbk@gmail.com

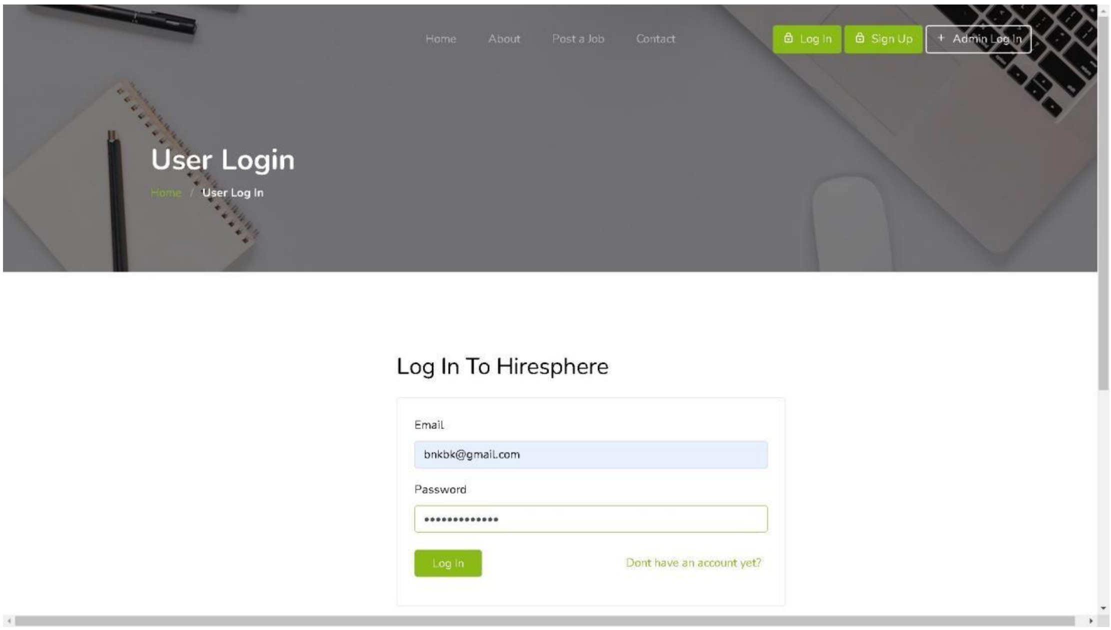
Password: ••••••••••••

Re-Type Password: •••••••••••••

Mobile Number: 7778888999

Upload Your Resume

User Login Page



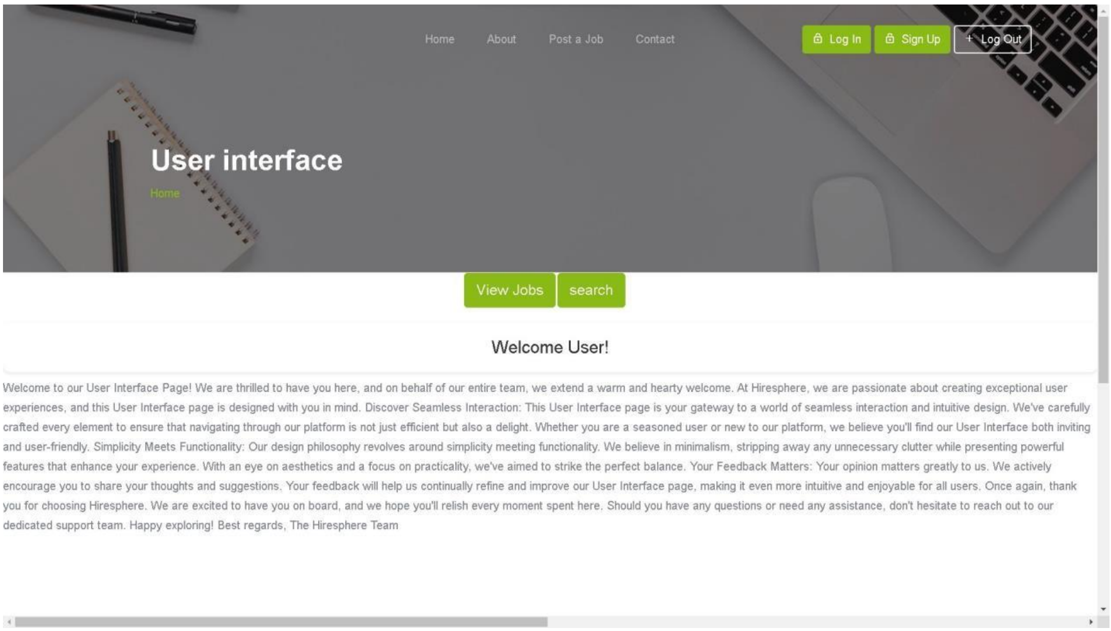
User Login

Log In To Hiresphere

Email: bnkbk@gmail.com

Password: •••••••••••••

User Interface

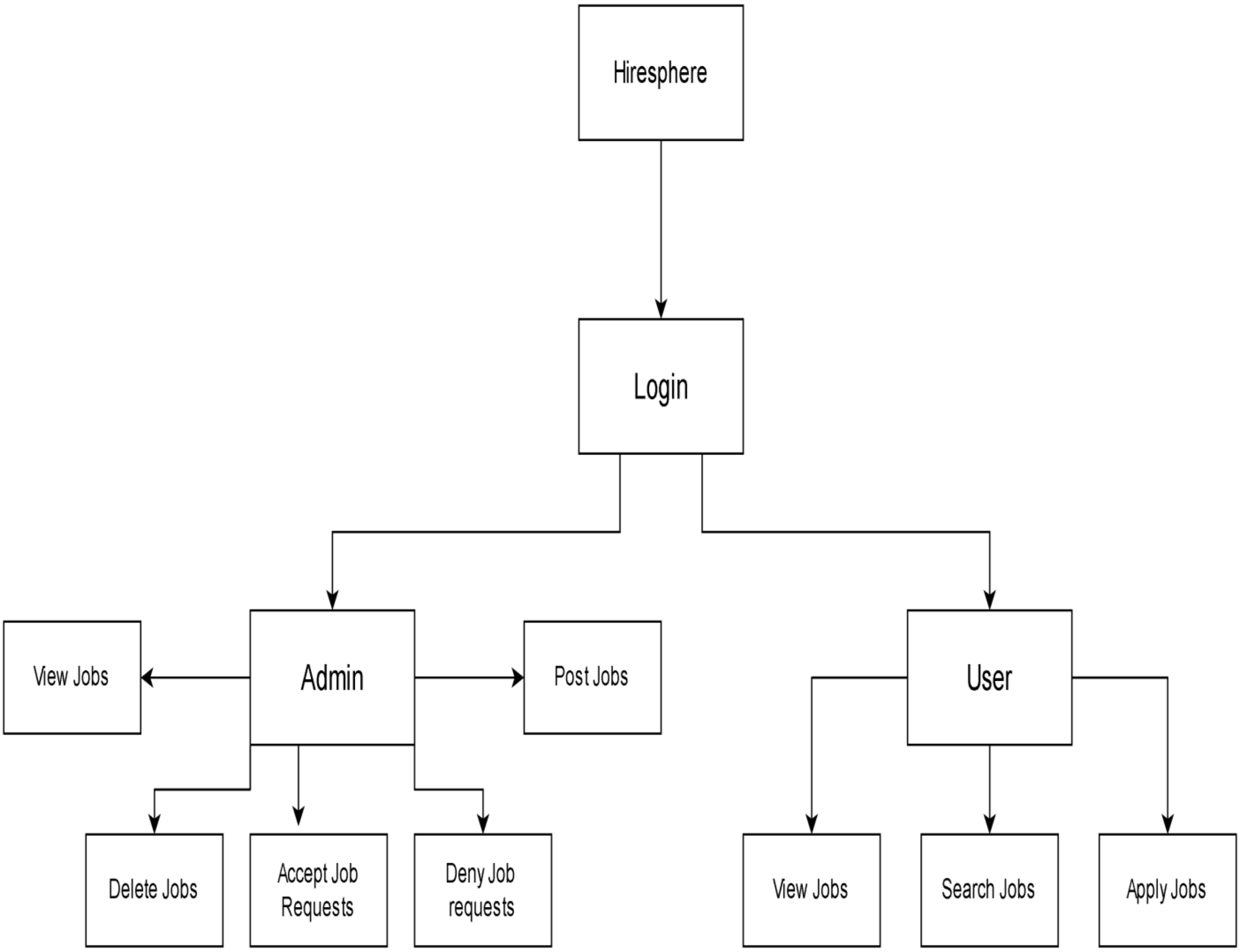


User interface

Welcome User!

Welcome to our User Interface Page! We are thrilled to have you here, and on behalf of our entire team, we extend a warm and hearty welcome. At Hiresphere, we are passionate about creating exceptional user experiences, and this User Interface page is designed with you in mind. Discover Seamless Interaction: This User Interface page is your gateway to a world of seamless interaction and intuitive design. We've carefully crafted every element to ensure that navigating through our platform is not just efficient but also a delight. Whether you are a seasoned user or new to our platform, we believe you'll find our User Interface both inviting and user-friendly. Simplicity Meets Functionality: Our design philosophy revolves around simplicity meeting functionality. We believe in minimalism, stripping away any unnecessary clutter while presenting powerful features that enhance your experience. With an eye on aesthetics and a focus on practicality, we've aimed to strike the perfect balance. Your Feedback Matters: Your opinion matters greatly to us. We actively encourage you to share your thoughts and suggestions. Your feedback will help us continually refine and improve our User Interface page, making it even more intuitive and enjoyable for all users. Once again, thank you for choosing Hiresphere. We are excited to have you on board, and we hope you'll relish every moment spent here. Should you have any questions or need any assistance, don't hesitate to reach out to our dedicated support team. Happy exploring! Best regards, The Hiresphere Team

MENU TREE


Hiresphere
Login
Admin
View Jobs
Post Jobs
Delete Jobs
Accept Job Requests
Deny Job requests
User
View Jobs
Search Jobs
Apply Jobs

MEETING MINUTES

Date:14/03/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Discussed about the topic

Date: 15/03/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Prepared synopsis for project

Date: 16/03/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Analysis of system requirements

Date: 20/03/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Started for frontend development class

Date: 23/03/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Started Javascript introduction class

Date: 27/03/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Started database connectivity &APIs class

Date: 22/05/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: started design of login page

Date: 25/05/23

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: started Documentation

Date: 29/05/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Design for project was prepared

Date: 31/05/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: started doing the modifications

Date: 05/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: completed modifications

Date: 08/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: started coding for user module

Date: 12/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: coding for Admin module

Date: 14/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: coding completed for Admin module

Date: 15/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Testing started for completed modules

Date: 16/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: User functionalities completed

Date: 19/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Designing of control module

Date: 19/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Designing of control module completed

Date: 22/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Documentation was continued

Date:26/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Documentation was continued

Date:28/06/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Documentation was continued

Date:12/07/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: coding completed

Date:13/07/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: started designing of data flow diagram

Date:25/07/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Designing of ER diagram

Date:01/08/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Designing of flowchart

Date:08/08/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: completion of updations.

Date:11/08/2023

Place: Christ Nagar College lab.

Members: Sarang S, Sidharth Sivan, Vikas S, Soorya Narayan R S

Discussion: Completion of the project

BIBLIOGRAPHY

BOOK REFERENCE

1. Rajib Mall, *"Fundamentals of software engineering"* , 2018, PHI Learning.
2. Shari Lawrence Pfleeger, *"Software engineering Theory and Practice",* 2010,Pearson.
3. Simon Holmes and Clive Harber, "*Getting MEAN with Mongo, Express, Angular, and Node"*, 2019, Manning Publications.
4. Chisholm, and May*, "Universal Design for Web Applications: Web Applications That Reach Everyone"*, O'Reilly Media, 2008

WEBSITE REFERENCE

1. https://geeksforgeeks.com/
2. https://www.smartworkonline.com/
3. https://nextjobfinder.com/
4. https://naukri.com/
5. https://naukri.com/