# **JobConnect: Platform For Jobs**

UCS2201 - Fundamentals and Practice of Software Development

### A PROJECT REPORT

### Submitted By

Saisandeep Sangeetham - 3122 22 5001 119 Shaun Allan H - 3122 22 5001 127 Shreyamanisha C Vinay - 3122 22 5001 130



Department of Computer Science and Engineering

Sri Sivasubramaniya Nadar College of Engineering
(An Autonomous Institution, Affiliated to Anna University)

Kalavakkam – 603110

July 2023

# Sri Sivasubramaniya Nadar College of Engineering (An Autonomous Institution, Affiliated to Anna University)

# **BONAFIDE CERTIFICATE**

Certified that this project report titled "JobConnect: Platform For		
Jobs" is the bonafide work of "Saisandeep Sangeetham (3122 22		
5001 119), Shaun Allan H (3122 22 5001 127),		
Shreyamanisha C Vinay (3122 22 5001 130) "who carried out		
the project work in the UCS2201 – Fundamentals and Practice of		
Software Development during the academic year 2022-23.		
Internal Examiner External Examiner		
Date:		

# **TABLE OF CONTENTS**

S.NO	CONTENTS	PG.NO
1	Introduction	1
2	Problem Statement	2
3	Exploration Beyond Problem Statement	3
4	Data Flow Diagrams (level 0, 1, 2)	4-6
5	Detailed Design	7-23
6	Description	24-29
7	Implementation	30-36
8	Test Cases	37-67

9	Limitations	68
10	Observations	69-70
11	Learning Outcomes	71
12	References	72

### 1. INTRODUCTION

The purpose of this report is to present the development and implementation of a job seeking platform called Job Connect using the C programming language. JobConnect aims to revolutionize the traditional job search process by providing a comprehensive platform for job seekers to connect with potential employers and explore employment opportunities and vice versa. This report will discuss the key features, design principles, and implementation details of JobConnect, highlighting the utilization of C programming concepts and techniques.

In today's highly competitive job market, finding suitable employment opportunities has become a challenging task for job seekers and similarly finding candidates that satisfy the specifications of a company has become difficult. To address this issue, the C project "JobConnect" has been developed as a job seeking platform, who's primary objective is to create a user-friendly platform that simplifies the job search process and enhances the overall experience for both job seekers and employers. By leveraging the power of C programming, the project aims to implement essential functionalities such as job listing, candidate profiles, application management, and communication channel that offers a streamlined and efficient approach to form a connection between the candidates and companies.

By leveraging C programming, the project demonstrates the versatility and power of the language in creating efficient and robust systems. Furthermore, this project serves as a learning opportunity for students to enhance their programming skills, particularly in C, and gain practical experience in developing real-world applications.

In the subsequent sections of this report, we will delve into the design and implementation aspects of JobConnect, showcasing the key functionalities, data structures, algorithms, and best practices employed in its development.

### 2. PROBLEM STATEMENT

Develop a software system to create a platform for Job Seekers and Talent Acquisition Managers where it acts as an intermediating device between the Job Seekers and Talent Acquisition Team.

Here, the Job Seekers can explore the available job positions which match their profile. And the Talent Acquisition Team can search for deserving applicants who will qualify their job requirements.

This can be done by calculating the recommendation percentage after inputting the constraints from both sides using the formula recommendation percentage is the ratio of accumulation of weighted scores of all the satisfying constraints to the weighted scores of all the available constraints multiplied by 100.

#### 3. EXPLORATION BEYOND PROBLEM STATEMENT

We have explored a lot beyond the problem statement. Here are the following:

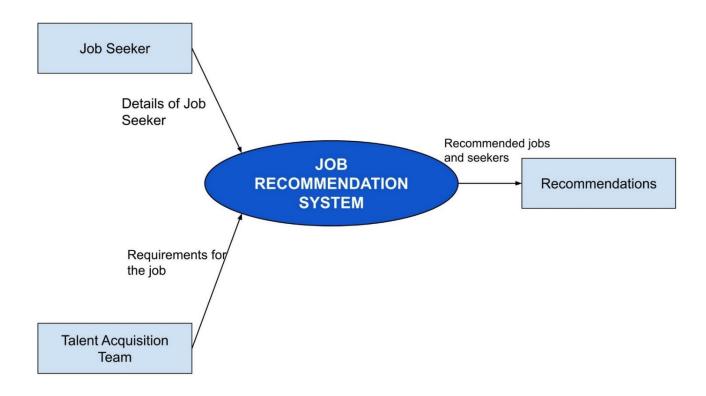
- Encrypted the password before storing in database
- Added more constraints (Job Seeker side Job Type, Company Type,
   Location, Day/Night Shift, Work from
   Home, Salary)

(Talent Acq side - Qualification, Age, Experience, Shift, CGPA, 10<sup>th</sup> & 12<sup>th</sup> Marks)

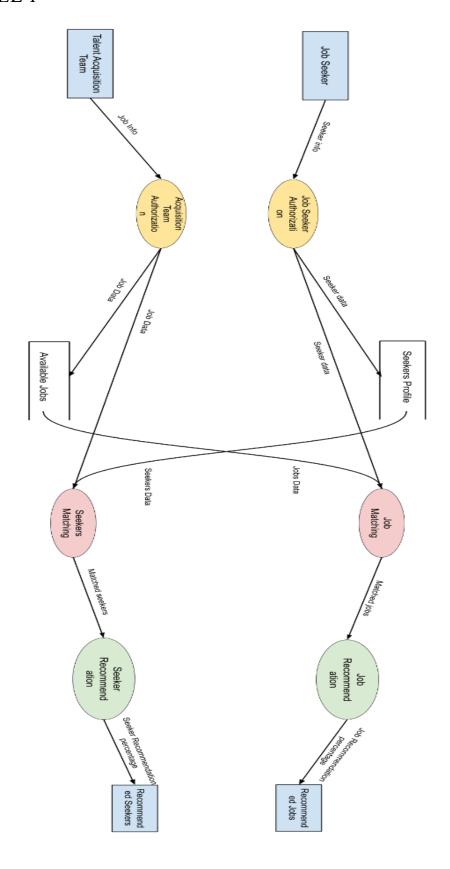
- Get the 10<sup>th</sup>, 12<sup>th</sup> and degree certificate from seeker to check manually
- Integrate Machine learning algorithms to calculate score
- Application status (Seeker applies a job to a company and the talent acquisition Side either rejects or accepts the application and this is notified to the seeker)

# 4. DATA FLOW DIAGRAMS

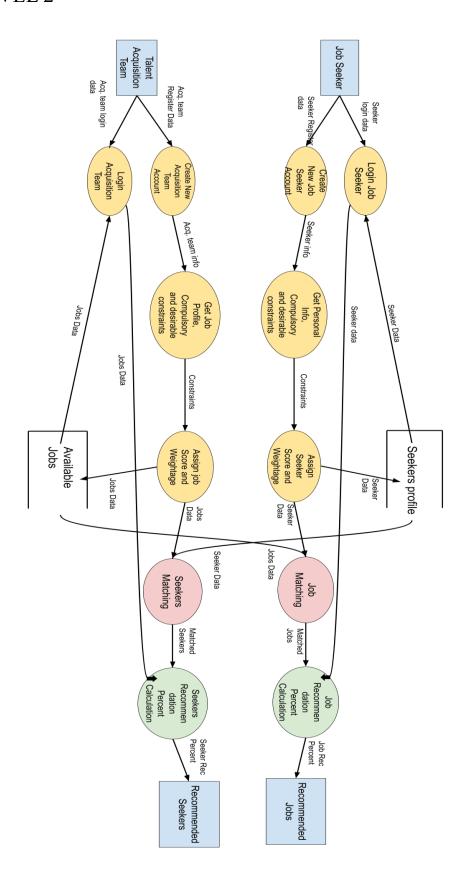
### 4.1. LEVEL 0



# 4.2. LEVEL 1

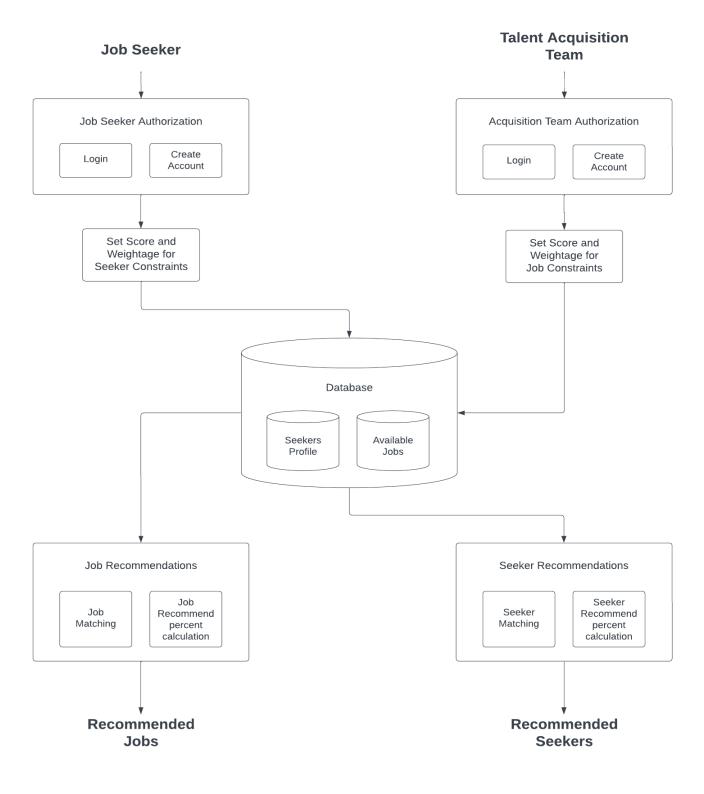


### 4.3. LEVEL 2



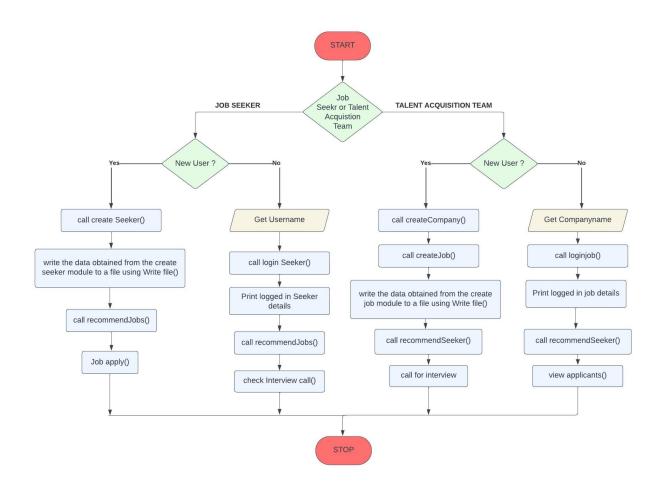
### 5. DETAILED DESIGN

# 5.1. ARCHITECTURAL DESIGN

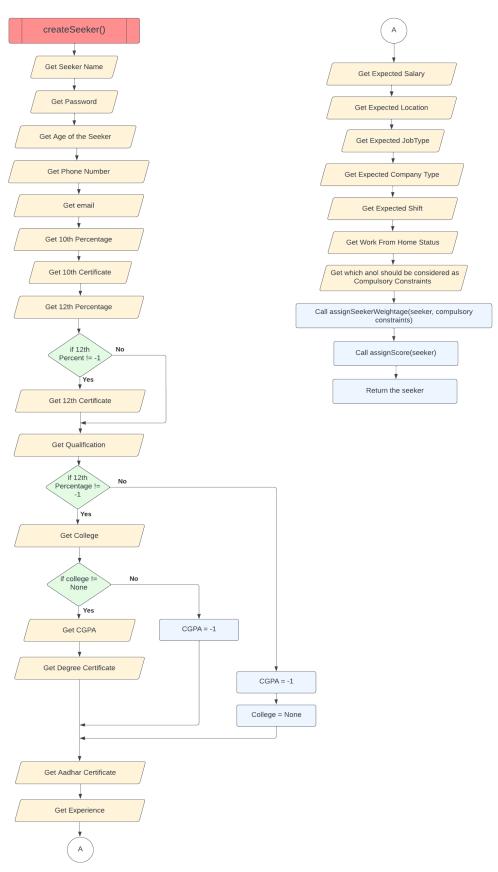


### 5.2. FLOWCHART FOR EACH MODULE

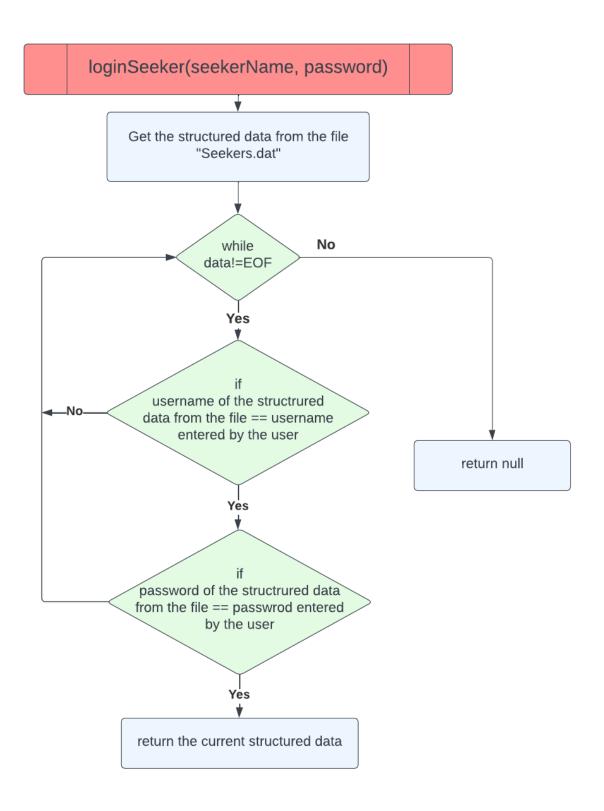
### 5.2.1. Main Module:



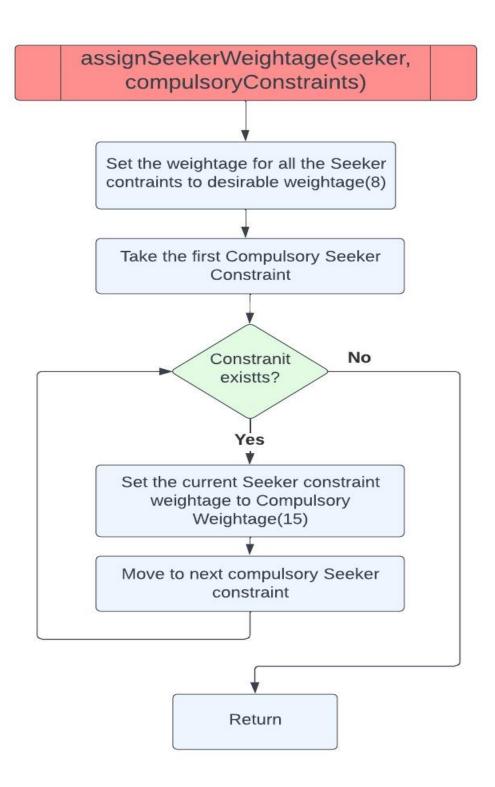
### 5.2.2. Create Seeker Module:



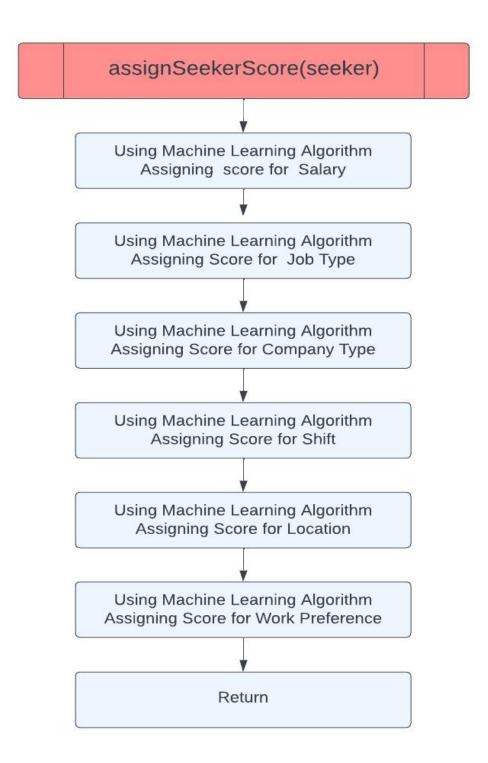
# 5.2.3. Seeker Login Module:



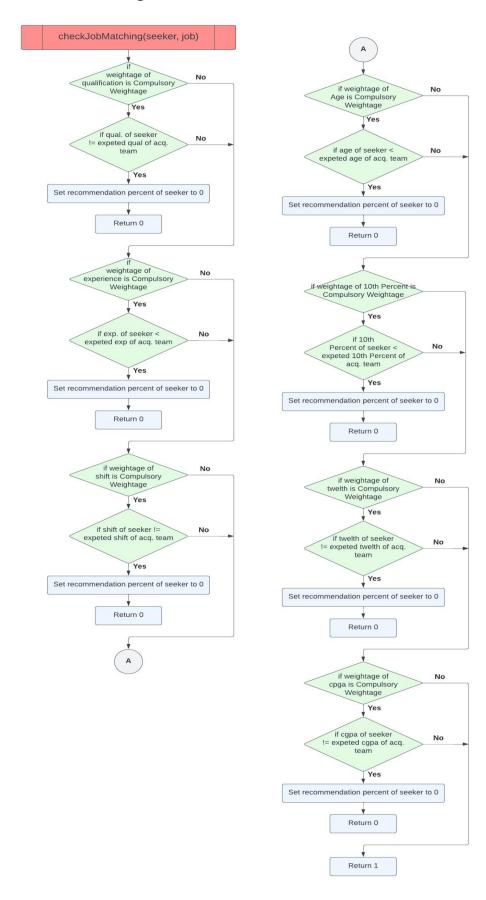
# 5.2.4. Assign Seeker Weightage:



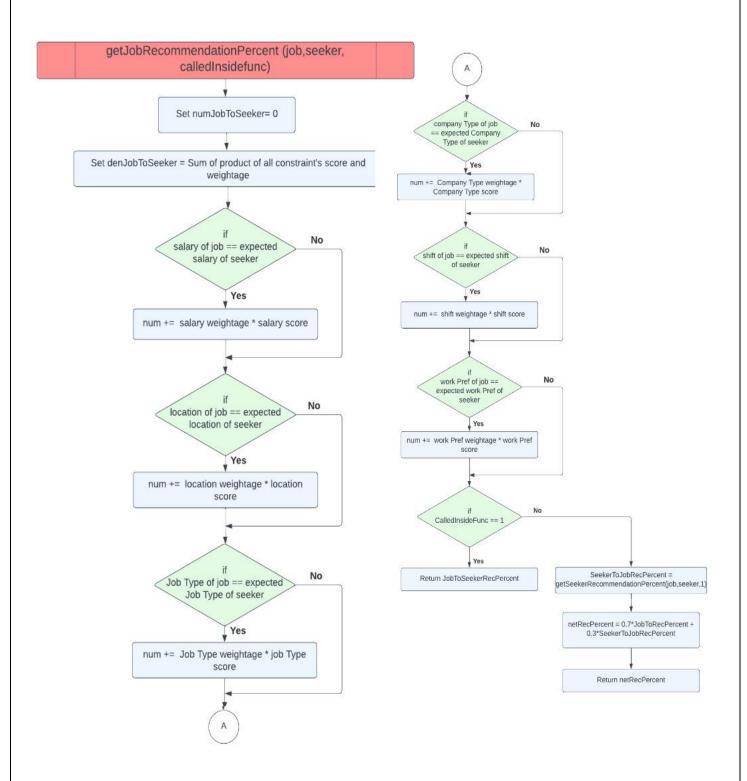
### 5.2.5. Assign Seeker Score:



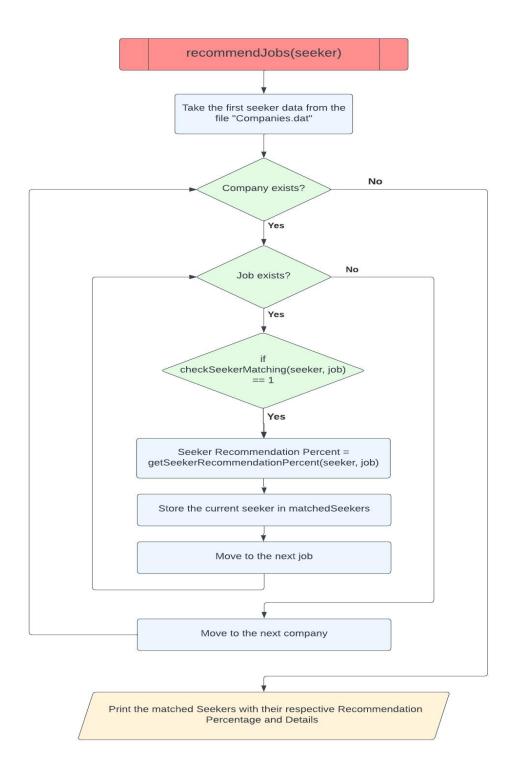
# 5.2.6. Check Job Matching:



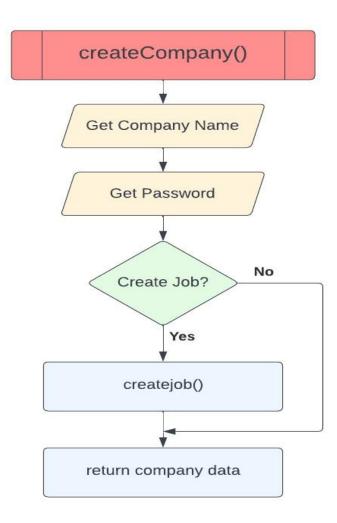
### 5.2.7. Get Job Recommendation Percent:



# 5.2.8. Recommend Jobs:



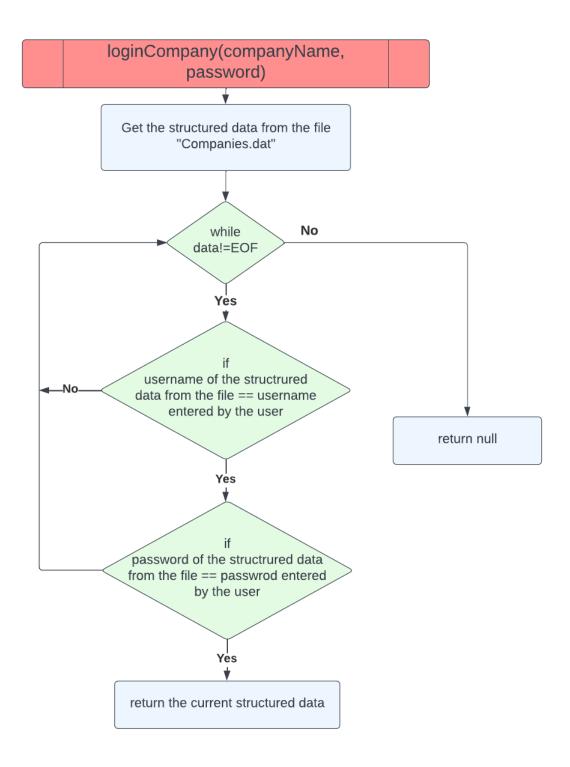
# 5.2.9. Create Company:



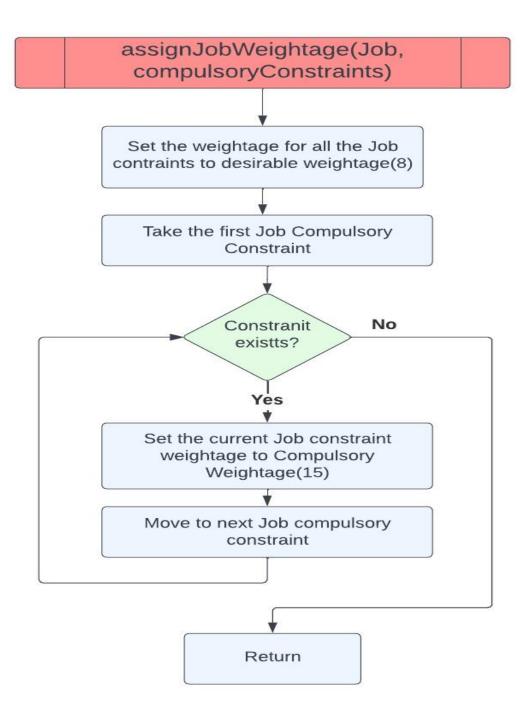
### 5.2.10. Create Job Module:



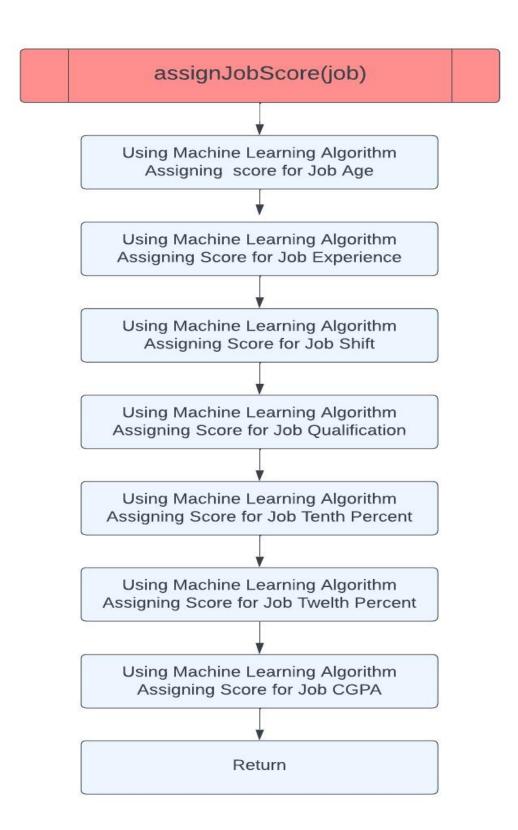
# 5.2.11. Job Login Module:



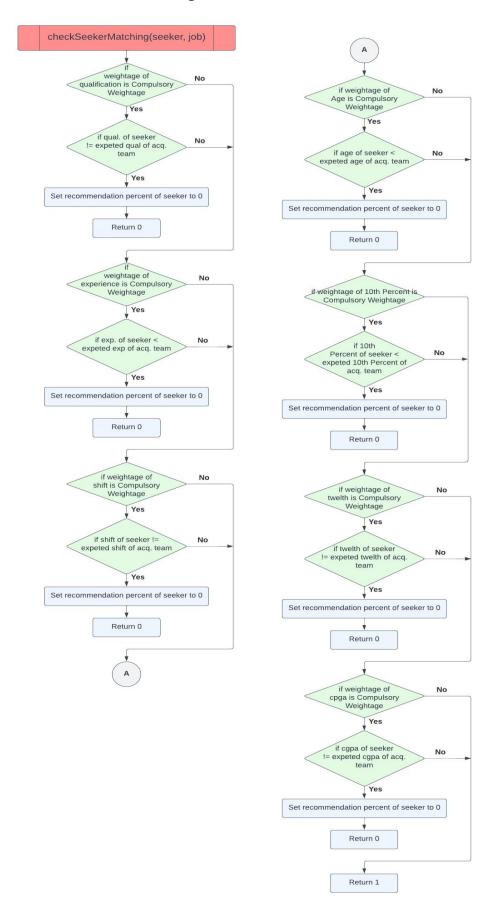
# 5.2.12. Assign Job Weightage:



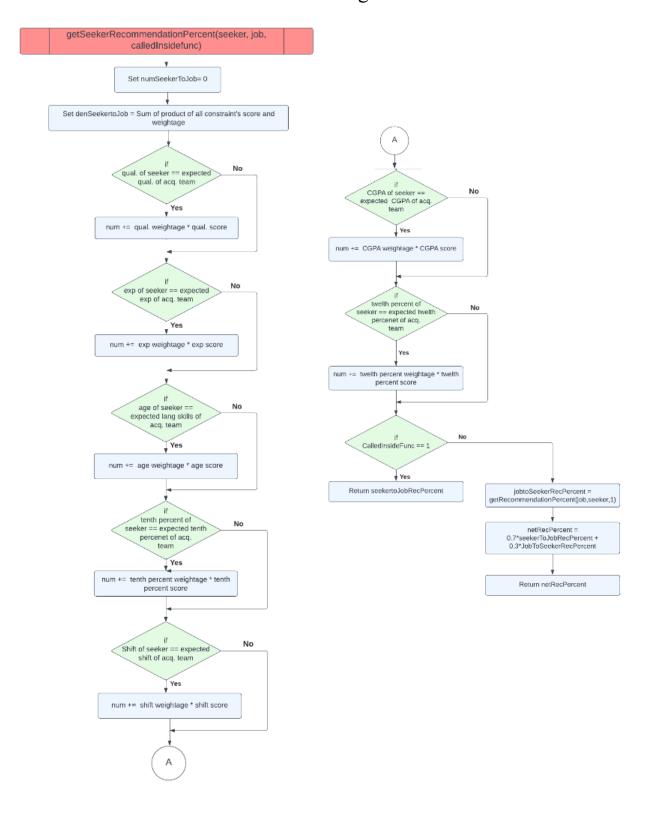
### 5.2.13. Assign Job Score:



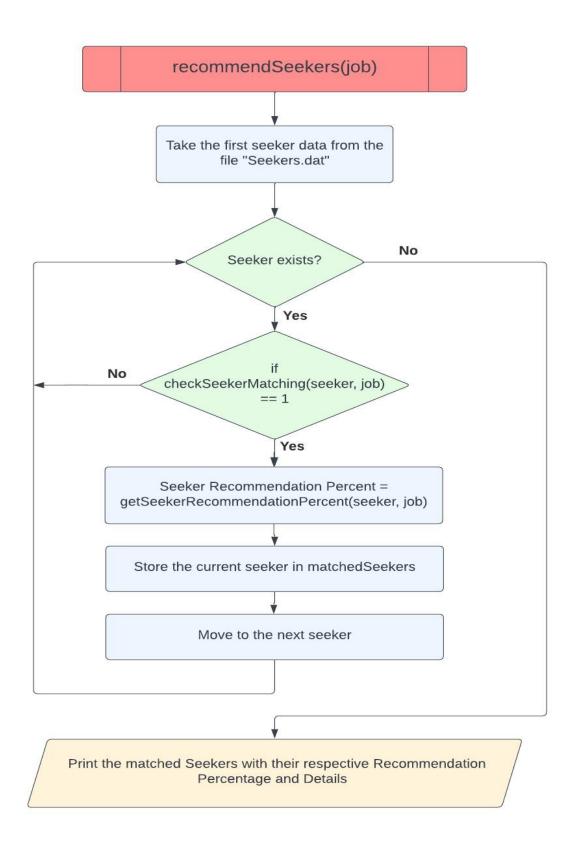
# 5.2.14. Check Seeker Matching:



# 5.2.15. Get Recommendation Percentage Module:



### 5.2.16. Recommend Seekers:



### 6. DESCRIPTION

#### 6.1. MAIN MODULE

- This main module serves as the central component that brings together all the functionalities and modules required for the system. It begins by prompting the user to choose between the job seeker or talent acquisition team options.
- For job seekers, it provides functionalities such as creating an account or logging in, with the data being stored in files. Additionally, seekers can access features like viewing recommended jobs and applying for them. An extra module is included specifically for the login side to handle checking for interview calls.
- On the other hand, for the talent acquisition team, the module supports functionalities such as creating a company profile, posting job openings, viewing recommended seekers, and initiating interview calls. Similarly, an extra module is available for the login side to facilitate viewing applicant information.
- By encompassing all these functionalities, the main module acts as the central hub for coordinating and integrating the different features required for both job seekers and the talent acquisition team

#### 6.2. CREATE SEEKER MODULE

• we simply ask the user for input regarding the constraints, such as seeker name, password, age, phone number, email, documents (10th, 12th), CGPA, qualification, etc. Next, we get information about the number of compulsory constraints and which constraints are required to be provided in SL.NO order. Finally, we call the assign weightage and assign score modules and return the seeker structure.

#### 6.3. SEEKER LOGIN MODULE

• The seeker name and password are passed through this module, and after getting the structured data from the file "Seekers.dat" and reading it from beginning to end, we check to see if the seeker name is available. If it is, we then check to see if the password is correct. If both constraints are satisfied, the module returns the seeker's current structured data, if neither of them fails, it returns null.

#### 6.4. ASSIGN SEEKER WEIGHTAGE MODULE

• In this module that receives an array of Compulsory Constraints as well as structured seeker data. Then, after first defining each constraint's weightage to a desirable value, we iterate over the compulsory constraint array, which contains the SL.NO, and in accordance with its contents, the weightage is changed to the value that represents the compulsory weightage value, returning null.

#### 6.5. ASSIGN SEEKER SCORE MODULE

• In this module's assignment, we used the pre-defined machine learning approach (KNN- algorithm) from the software that contains the training data. With the help of this, we automatically determine a score based on the preferences of the seeker and, in the end, determine the score for those constraints.

#### 6.6. CHECK JOB MATCHING MODULE

• This module passes structured data that involves the job and seeker, and then checks, if a constraint is given as a compulsory, then it is checked to see if the expected seeker constraint is different from the expected constraint of the talent acquisition team. If both conditions are satisfied, then the recommendation percent will be set to 0, but if any of them fails, the module will move on to the next constraint.

#### 6.7.GET JOB RECOMMENDATION PERCENT MODULE

- In this module, the recommendation percentage is calculated based on a set of constraints between job requirements and seeker preferences. The calculation begins by initializing the numerator to 0 and the denominator to the sum of the products of all constraint scores and their corresponding weightages.
- The module then iterates through each constraint, comparing the expected constraint of the job to the expected constraint of the seeker. If the constraint is satisfied, the numerator is incremented by the product of the constraint weightage and its score. This process is repeated for all the constraints until they have all been evaluated.
- After evaluating the constraints, the module checks if the value returned by the function "call within fun" is equal to 1. If so, the job is returned to the seeker with the recommended percentage. Alternatively, if the function call is not equal to 1, the seeker to job recommendation percentage is calculated using the "get seeker recommendation percent" module.
- Finally, the net recommendation percentage is computed as 0.7 times the job to seeker recommendation percentage plus 0.3 times the seeker to job recommendation percentage. This net recommendation percentage is then returned as the result of the module.

#### 6.8. RECOMMEND JOBS MODULE

• The structured data of the seeker is passed as an input to this module's recommend jobs; this module now also accepts data from the "companies.dat". Once the dat file has been extracted, each company is iterated through, first checking to see if they exist. If they do, then the internally matched jobs are incremented the job matching module is then called, and if the returned value is 1, the job recommendation percent is calculated and the current job's data is stored in the matched jobs array. examining the array of matched jobs, print the recommended percentage for that job, and then go on to the next job. At last, then go on to the next company.

#### 6.9. CREATE COMPANY MODULE

• In this module, the inner structure that holds the job structure has a maximum of 15 jobs and is called Create Company. where we provide the company's name, password, and if they want to generate jobs before calling the create jobs module and, finally, return the company data.

#### 6.10. CREATE JOB MODULE

• We simply require data from the user regarding the constraints, such as the job title, location, job kind, business type, anticipated age, qualification, experience, Exp CGPA, etc. Next, we get information about the number of compulsory constraints and which constraints are required to be provided in SL.NO order. Finally, we call the assign weightage and assign score modules and return the Job structure.

#### 6.11.JOB LOGIN MODULE

• The Job name and password are passed through this module, and after getting the structured data from the file "Companies.dat" and reading it from beginning to end, we check to see if the Job name is available. If it is, we then check to if the password is correct. If both constraints are satisfied, the module returns the job's current structured data, if neither of them fails, it returns null.

#### 6.12. ASSIGN JOB WEIGHTAGE MODULE

• In this module that receives an array of Compulsory Constraints as well as structured job data. Then, after first defining each constraint's weightage to a desirable value, we iterate over the compulsory constraint array, which contains the SL.NO, and in accordance with its contents, the weightage is changed to the value that represents the compulsory weightage value, returning null.

#### 6.13. ASSIGN JOB SCORE MODULE

• In this module's assignment, we used the pre-defined machine learning approach (KNN- algorithm) from the software that contains the training data. With the help of this, we automatically determine a score based on the preferences of the talent acquisition team and, in the end, determine the score for those constraints.

#### 6.14. CHECK SEEKER MATCHING MODULE

• This module passes structured data that involves the job and seeker, and then checks, if a constraint is given as a compulsory, then it is checked to see if the expected job constraint is different from the expected constraint of the seeker. If both conditions are satisfied, then the recommendation percent will be set to 0, but if any of them fails, the module will move on to the next constraint.

#### 6.15.GET SEEKER RECOMMENDATION PERCENT MODULE

- In this module, the recommendation percentage is calculated based on a set of constraints between seeker requirements and job preferences. The calculation begins by initializing the numerator to 0 and the denominator to the sum of the products of all constraint scores and their corresponding weightages.
- The module then iterates through each constraint, comparing the expected constraint of the seeker to the expected constraint of the job. If the constraint is satisfied, the numerator is incremented by the product of the constraint weightage and its score. This process is repeated for all the constraints until they have all been evaluated.
- After evaluating the constraints, the module checks if the value returned by the function "call within fun" is equal to 1. If so, the seeker is returned to the job with the recommended percentage. Alternatively, if the function call is not equal to 1, the job to seeker recommendation percentage is calculated using the "get job recommendation percent" module.
- Finally, the net recommendation percentage is computed as 0.7 times the seeker to job recommendation percentage plus 0.3 times the job to seeker

recommendation percentage. This net recommendation percentage is then returned as the result of the module.

#### 6.16. RECOMMEND SEEKERS MODULE

• The structured data of the job is passed as an input to this module's recommend seekers; this module now also accepts data from the "Seekers.dat". Once the dat file has been extracted, each seeker is iterated through, first checking to see if they exist. If they do, the seeker matching module is then called, and if the returned value is 1, the seeker recommendation percent is calculated and the current seeker's data is stored in the matched seekers array. examining the array of matched seekers, print the recommended percentage for that job, and then go on to the next seeker.

# 7. IMPLEMENTATION

# 7.1. DATA ORGANIZATION

### 7.1.1. SEEKER

Construct Used: Structures stored in Files

Contents of the Structure - Seeker:

Data	Data Type
Seeker Name	String
Password	String
Age	Integer
Phone Number	Long Long Integer
Email	String
Qualification	String
Experience	Integer
College	String
Tenth Percentage	Float
Twelfth Percentage	Float
CGPA	Float
Expected Salary	Integer
Salary Score and Weightage	Integer
Expected Location	String
Location Score and Weightage	Integer
Expected Job Type	String
Job Type Score and Weightage	Integer
Expected Company Type	String
Company Type Score and Weightage	Integer
Expected Shift	Character
Shift Score and Weightage	Integer
Work From Home Status	Character
Work From Home Score and	Integer
Weightage	
No of Applied Jobs	Integer
No of Newly Appeared Jobs	Integer
Applied Jobs	Array of AppliedJobs – User Defined Structure

Reason for Choosing this type of Construct:

Seeker is basically a collection of heterogenous data (dissimilar data types). So, we use a structure to store a seeker. After getting the details of the seeker from the user, we store them in Seekers.dat file to store the data permanently.

### 7.1.2. APPLIED JOBS

Construct Used: Structures

Contents of the Structure – Applied jobs:

Data	Data Type
Company Name	String
Job	String
Status	Integer

Reason for choosing this type of Construct:

Applied jobs is used to store the information of the jobs that the seeker has applied for. So, we see, it stores heterogenous data. Therefore, we use structures.

#### **7.1.3. COMPANY**

Construct Used: Structures stored in Files

Contents of the Structure – Company:

Data	Data Type
Company Name	String
Password	String
No of Jobs	Integer
Jobs	Array of Job – User defined Structure

### Reason for Choosing this type of Construct:

Company will contain a collection of data related to the company. A Company will contain jobs inside it which is another user defined structure. We see the data for a Company is heterogenous. So, structures are the best suited to store this type of data. After getting the data for structures, we store the structure in Companies.dat file to store them permanently.

### 7.1.4. JOB

Construct Used: Structure

Contents of the Structure – Job:

Data	Data Type
Job	String
Salary	Integer
Location	String
Job Type	String
Company Type	String
Work From Home Status	Character
Shift	Character
Shift Score and Weightage	Integer
Expected Age	Integer
Age Score and Weightage	Integer
Expected Experience	Integer
Experience Score and Weightage	Integer
Expected Qualification	String
Qualification Score and Weightage	Integer
Expected Tenth Percentage	Float
Tenth Percent Score and Weightage	Integer
Expected Twelfth Percentage	Float
Twelfth Percent Score and Weightage	Integer
Expected CGPA	Float
CGPA Score and Weightage	Integer

Reason for choosing this type of Construct:

Jobs is used to store the information of jobs under a company. We have so many data regarding jobs that are of dissimilar data type. Therefore, it is best suited to use structures to store the information of the job.

#### 7.1.5. COMPANY APPLICATION

Construct Used: Structures stores in Files

Contents of the Structure – Company Application:

Data	Data Type
Company	String
No of Applicants	Integer
Newly Appeared Applicants	Integer
Applications	Array of Applications – User Defined Structure

Reason for choosing this type of Construct:

A Company Application is a collection of data which stores the data of applications for a particular company. These data are heterogenous. Therefore, structures can be used to store the data. After storing the information in the structure, we store the structure in Applications.dat to store the data permanently.

#### 7.1.6. APPLICATION

Construct Used: Structures

Contents of the structure – Application:

Data	Data Type
Applicant	String
Job	String

Reason for choosing this type of Construct:

Applications stores the information of the Applicant that is the applicant Name and the Job he has applied for. Though the data are homogenous, it is convenient to store and access data if we store them in a structure.

#### 7.2. EXTERNAL LIBRARIES AND APIS

used to hash a string in SHA256 hashing algorithm using the SHA256() function of the library in which the string to be hashed is passed as a parameter

score.h -- contains user modified internet-based machine learning functions for KNN Algorithm

#### 7.3. USER INTERFACE DESIGN

The project is an application that runs on the terminal. To provide a good look and feel to the users, the outputs of the program is well formatted. Also, to provide a good flow or to avoid confusion to the users, we refresh the terminal, that is we remove the previous module's contents from the terminal and display only the current module's contents.

#### Home Page:

```
Are you a

=> 1. Job Seeker

=> 2. Talent Acquisition Manager

Enter 1 or 2: 1

Do you want to

=> 1.Create a new Seeker account

=> 2.Login

Enter -1 to go back
Enter 1, 2 or -1: 2
```

#### Login Page:

The contents of the home page are removed and only the contents of login Seeker module are displayed

/ · · · ·	PLATFORMS USED FOR CODE DEVELOPMENT
	VS Code was used to develop and implement the Project using C language. Outputs were also recorded in the terminal of VS Code.

### 8. TEST CASES

#### 8.1. MAIN MENU

## 8.1.1. Are you a Job Seeker or Talent Acq. Manager

Input	Expected Output	Actual Output	Validation
Any sort of string input	Warning	Are you a  => 1. Job Seeker  => 2. Talent Acquisition Manager  Enter 1 or 2: jdn Invalid Input. Enter 1 or 2  Are you a  => 1. Job Seeker  => 2. Talent Acquisition Manager  Enter 1 or 2:	Successful
Integer other than 1 or 2	Warning Prompt: Invalid Input	Are you a  => 1. Job Seeker => 2. Talent Acquisition Manager  Enter 1 or 2: 5 Invalid Input.Please Enter 1 or 2  Are you a => 1. Job Seeker => 2. Talent Acquisition Manager  Enter 1 or 2:   Enter 1 or 2:   ### Insulation Manager  ### Insulatio	Successful
1	Move to Job Seeker page	Are you a  => 1. Job Seeker => 2. Talent Acquisition Manager  Enter 1 or 2: 1  Do you want to => 1.Create a new Seeker account => 2.Login  Enter -1 to go back Enter 1, 2 or -1: ■	Successful

2	Move to	Are you a => 1. Job Seeker	Successful
	Company Page	=> 1. Job Seeker => 2. Talent Acquisition Manager	
		Enter 1 or 2: 2	
		Do you want to => 1.Create a new Company account => 2.Login	
		Enter -1 to go back Enter 1,2 or -1: ■	

## 8.2. SEEKER MENU

# 8.2.1. Create Account or Login

Input	Expected	Actual Output	Validation
	Output		
Any sort of string input	Warning Prompt: Invalid Input	Do you want to  => 1.Create a new Seeker account => 2.Login  Enter -1 to go back Enter 1, 2 or -1: jxcd Invalid Input. Enter 1,2 or -1	Successful
		Do you want to  => 1.Create a new Seeker account  => 2.Login  Enter -1 to go back Enter 1, 2 or -1:	

Integer other than 1, 2, or -1	Prompt: Invalid	Do you want to  => 1.Create a new Seeker account => 2.Login  Enter -1 to go back Enter 1, 2 or -1: 6 Invalid Input. Enter 1,2 or -1	Successful
		Do you want to  => 1.Create a new Seeker account => 2.Login  Enter -1 to go back Enter 1, 2 or -1:	
-1	Move to main menu	Are you a  => 1. Job Seeker  => 2. Talent Acquisition Manager  Enter 1 or 2:	Successful
1	Move to create seeker module	Create A New Seeker Account Enter Name:	Successful
2	Move to login Seeker Module	Enter -1 to go back Enter username: ■	Successful

## 8.3. CREATE SEEKER

## 8.3.1. Seeker Name

Expecte	Actual Output	Validatio
d		n
Output		
Seeker	Enter password: pras	Successful
	Password length must be between 8 - 15	
•		
_	Enter password: psanathsanthjbshvdhvsfff	
	Password length must be between 8 - 15	
_	Enter password:	
msteau?		
Seeker	Enton naccinand, DDACANNA	Successful
Name is	Password must have atleast 1 lowercase character	
stored		
	Enter password:	
	d Output Seeker Name Already Exists. Do you want to login Instead?  Seeker Name is	Output  Seeker Name Already Exists. Do you want to login Instead?  Seeker Name is  Enter password: pras Password length must be between 8 - 15  Enter password: psanathsanthjbshvdhvsfff Password length must be between 8 - 15  Enter password: psanathsanthjbshvdhvsfff Password length must be between 8 - 15  Enter password: pras Password: psanathsanthjbshvdhvsfff Password length must be between 8 - 15  Enter password: psanathsanthjbshvdhvsfff Password length must be between 8 - 15

# 8.3.2. Password

Input	Expected Output	Actual Output	Validation
Less than 8 character s or more than 15 character s	Warning Prompt: Password length must be between 8 - 15	Enter password: pras Password length must be between 8 - 15  Enter password: psanathsanthjbshvdhvsfff Password length must be between 8 - 15  Enter password:	Successful
No Lowercas e character s	Warning Prompt: Password must have at least 1 lowercase character.	Enter password: PRASANNA Password must have atleast 1 lowercase character  Enter password:	Successful
No Uppercas e character	Warning Prompt: Password must have at least 1 uppercase character	Enter password: prasanna Password must have atleast 1 uppercase character  Enter password:	Successful
No Digits	Warning Prompt: Password must have at least 1 digit	Enter password: Prasanna Password must have atleast 1 digit  Enter password:	Successful

Spaces	Warning Prompt: Password cannot contain spaces	Enter password: Prasanna 1 Password cannot contain spaces Enter password:	Successful
No Special Character s	Warning Prompt: Password must have at least 1 special character	Enter password: Prasanna1 Password must have atleast 1 special charact Enter password:	Successful
All the constrain ts satisfied	Password accepted	Enter password: Prasanna@1  Confirm Password:	Successful

## 8.3.3. Confirm Password

Input	Expected	Actual Output	Validation
	Output		
Confirm	Warning	Enter password: Prasanna@1	Successful
Password!	Prompt:		
=	Passwords	Confirm Password: kjsdbg	
Password	do not	Passwords do not match	
	match		
		Confirm Password:	
Confirm	Password	Confirm Password: Prasanna@1	Successful
Password	gets	CONTINII PASSWOLU. PLASAIIIA@I	
==	stored		
Password		Enter Age(15-70):	

# 8.3.4. Age

Input	<b>Expected Output</b>	Actual Output	Validation
Any sort of string input	Warning Prompt: Invalid Input. Age cannot contain an alphabet or special character.	Enter Age(15-70): kjxcb Invalid Input. Age cannot contain an alphabet Enter Age(15-70):	Successful
Integer Less than 15 or greater than 70	Warning Prompt: Your age is too young or old for an employee . A valid age range between 15 - 70	Enter Age(15-70): 12 Your age is too young for an employee. A valid a  Enter Age(15-70): 76 Your age is too old for an employee. A valid age	Successful
Negativ e Integer	Warning Prompt: Invalid Input. Age cannot be negative.	Enter Age(15-70): -18 Invalid Input. Age cannot be neagtive  Enter Age(15-70):	Successful

Integer between	Age is stored	3	Enter Age(15-70): 20	Successful
15 - 70				
			Enter Phone Number:	

## 8.3.5. Phone Number

Input	Expected	Actual Output	Validatio
	Output		n
Any sort of string input	Warning Prompt: Invalid Input. Age cannot contain an alphabet or special character.	Enter Phone Number: sdhv Invalid Phone Number. Phone number cannot contain alphabets  Enter Phone Number:	Successful
Not 10 digits	Warning Prompt: Your age is too young or old for an employee . A valid age range between 15 - 70	Enter Phone Number: 62386535 Invalid Phone Number. Please enter a 10 digit Phone number.  Enter Phone Number:	Successful

Any 10-	Phone		Not
digit	Number	Enter Phone Number: 1234567890	Successful
Number	is stored.	Effect Thore Number: 1254507050	
. Even if			
its a		Enter email Id:	
non-		<del>_</del>	
existent			
phone			
number			

#### 8.3.6. Email Id

Input	Expected	Actual Output	Validation
	Output		
Any	Email gets		Not
form of	stored	Enter email Id: prasanth	Successful
string		Erreer email far prasarren	
even if			
it's not		Enter 10th Percentage:	
a valid		Litter 10th Fer telltage.	
mail id			

# 8.3.7. 10th Percentage

Input	Expecte	Actual Output	Validatio
	d Output		n
Any sort of string input	Warning Prompt: Invalid Input. 10th Percenta ge cannot contain alphabets or special character s	Enter 10th Percentage: kjxcb Invalid Input. 10th Percentage cannot contain  Enter 10th Percentage: ■	Successful
Float value greater than 100	Warning Prompt: Invalid Input. 10th Percenta ge cannot be greater than 100	Enter 10th Percentage: 150.7 Invalid Input. 10th Percentage cannot be greater than 100.  Enter 10th Percentage:	Successful
Negativ e Float value	Warning Prompt: Invalid Input. 10 <sup>th</sup> Percent cannot be negative.	Enter 10th Percentage: -5.6 Invalid Input. 10th Percentage cannot be neagti Enter 10th Percentage:	Successful

Float	10th	Enter 10th Percentage: 67.8	Successf
value	percent is		ul
betwee	stored	Enter location of 10th Certificate (*PDF ONLY):	
n 0 to		Effect location of loth ect ciritate ( 151 one);	
100			

## 8.3.8. 10th Certificate Location

Input	Expecte	Actual Output	Validatio
	d Output		n
Non pdf documen ts	Warning Prompt: Invalid file format. Please upload a PDF file	Enter location of 10th Certificate (*PDF ONLY): C:\Users Invalid file format. Please upload a PDF file  Enter location of 10th Certificate (*PDF ONLY):	Successful
File is not present in that location	Warning Prompt: Invalid file location	Enter location of 10th Certificate (*PDF ONLY): C:\Users Invalid file format. Please upload a PDF file  Enter location of 10th Certificate (*PDF ONLY):	Successful

Pdf	Pdf	Total lastin of 40th Cartificate (*DDT 041)(). C.\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	Successf
documen	documen	Enter location of 10th Certificate (*PDF ONLY): C:\Users\shaun\Do	ul
t with	t is stored	_	
correct	in the	Enter 12th Percentage (Enter -1 if not attended):	
file	documen		
location	ts folder.		

# 8.3.9. 12th Percentage

Input	Expected Output	Actual Output	Validatio n
Any sort of string input	Warning	Enter 12th Percentage (Enter -1 if not attended): ssm Invalid Input. 12th Percentage cannot contain alphabe  Enter 12th Percentage (Enter -1 if not attended):	Successf
Float value greater than 100	Warning Prompt: Invalid Input. 12th Percentage cannot be greater than 100	Enter 12th Percentage (Enter -1 if not attended): 156.7 Invalid Input. 12th Percentage cannot be greater than 100 Enter 12th Percentage (Enter -1 if not attended):	Successful

-1	12th Certificate is not asked, College and CGPA is automatical ly set as None and -1	Enter 12th Percentage (Enter -1 if not attended): -1  Enter Qualification: 10th  Enter location of Aadhar Card (*PDF ONLY): ■	Successful
Negativ e Float value	Warning Prompt: Invalid Input. 12 <sup>th</sup> Percent cannot be negative.	Enter 12th Percentage (Enter -1 if not attended): -3.4 Invalid Input. 12th Percentage cannot be neagtive  Enter 12th Percentage (Enter -1 if not attended):	Successful
Float value betwee n 0 to 100	12th percent is stored	Enter 12th Percentage (Enter -1 if not attended): 67.8  Enter location of 12th Certificate (*PDF ONLY):	Successful

# 8.3.10. 12th Certificate Location \*Same as 10th Certificate Location (7.3.7)

## 8.3.11. Qualification

Input	Expected	Actual Output	Validatio
	Output		n
Other than the qual. that are specied in the program	Warning Prompt: Invalid Qualificatio n. Please enter again	Enter Qualification: bca Invalid Qualification. Please enter again  Enter Qualification:	Successfu 1
Qual. that are specifie d in our program	Qualificatio n is stored	Enter Qualification: be  Enter College Name(None if not attended):	Successfu 1

NOTE: The characters in the qualification are automatically converted into uppercase characters

# 8.3.12. College

Input	Expected	Actual Output	Validati
	Output		on
None	CGPA is automatical ly taken as - 1 and degree certificate is not asked	Enter College Name(None if not attended): none  Enter location of Aadhar Card (*PDF ONLY):	Successful
Any string even if it is not an actual colleg e	College is stores	Enter College Name(None if not attended): ssn college of engi Enter College CGPA (Enter -1 if not attended): ■	Not Successf ul

NOTE: The first letter of each word is automatically converted into uppercase and the rest into lowercase characters

## 8.3.13. CGPA

Input	Expecte d Output	Actual Output	Validati on
Any sort of string input	Warning Prompt: Invalid Input. CGPA cannot contain alphabet s or special characte rs	Enter College CGPA (Enter -1 if not attended): jhvf Invalid Input. CGPA cannot contain alphabets or special charac  Enter College CGPA (Enter -1 if not attended): -5	Successful
Float value greater than 10	Warning Prompt: Invalid Input. CGPA cannot be greater than 10	Enter College CGPA (Enter -1 if not attended): 1 Invalid Input. 12th cannot be greater than 10  Enter College CGPA (Enter -1 if not attended):	Successful
Negati ve Float value	Warning Prompt: Invalid Input. CGPA cannot be negative	Enter College CGPA (Enter -1 if not attended): -5 Invalid Input. CGPA cannot be neagtive  Enter College CGPA (Enter -1 if not attended): 8 7	Successful

Float	CGPA is		Successf
value	stored	Enter College CGPA (Enter -1 if not attended): 8.7	ul
betwee			
n 0 to		Enter location of Degree Certificate (*PDF ONLY):	
10		enter rotation of begree eer efficate ( 151 one).	

## 8.3.14. Degree Certificate Location

\*Same as 10<sup>th</sup> Certificate Location (7.3.7)

#### 8.3.15. AADHAR Certificate Location

\*Same as 10<sup>th</sup> Certificate Location (7.3.7)

## 8.3.16. Experience

Input	Expected	Actual Output	Validatio
	Output		n
Any	Warning		Successf
form of	Prompt:	Enter Experience in years: xhhd	ul
string	Invalid	Invalid Input. Experience cannot contain an alphabet or	
	Input.		
	Experien	Enter Experience in years:	
	ce cannot	· · · · · · · ·	
	contain		
	an		
	alphabet		
	or special		
	character		

Negativ e Integer	Invalid Input. Experien	Enter Experience in years: -1 Invalid Input. Experience cannot be neagtive	Successf ul
	ce cannot be negative	Enter Experience in years:	
Integer	Experien		Successf
greater than age	ce is greater than Age. Please enter a valid experienc e.	Enter Experience in years: 40 Experience is greater than Age. Please enter a valid experience Enter Experience in years:	ul
Integer	Experien		Successf
betwee	ce is	Enter Experience in years: 5	ul
n 0 -	stored		
Age		Enter Expected Monthly Salary(Rs.5000 - Rs.10	

# 8.3.17. Salary

Input	Expected	Actual Output	Validatio
	Output		n
Any	Warning		Successfu
form of	Prompt:	Enter Expected Monthly Salary(Rs.5000 - Rs.1000000): xjk	1
string	Invalid	Invalid Input. Salary cannot contain an alphabet or special ch	
	Input.		
	Salary	Enter Expected Monthly Salary(Rs.5000 - Rs.1000000):	
	cannot		
	contain an		
	alphabet		
	or special		
	character		

Integer less than 5000 or More than 100000 0	Invalid Input. Please enter a salary between in the range Rs.5000 - Rs.100000	Enter Expected Monthly Salary(Rs.5000 - Rs.1000000): 400 Invalid Input. Please enter a salary between in the range Rs.500 Enter Expected Monthly Salary(Rs.5000 - Rs.1000000): 10000000000 Invalid Input. Please enter a salary between in the range Rs.500 Enter Expected Monthly Salary(Rs.5000 - Rs.1000000):	Successfu 1
Integer betwee n 5000 - 100000 0	Salary is stored.	Enter Expected Monthly Salary(Rs.5000 - Rs.1000000): 1000000  Enter expected location of workplace:	Successfu 1

#### 8.3.18. Job Location

9. Input	Expected	Actual Output	Validation
	Output		
Other than	Warning		Successful
the cities	Prompt:	Enter expected location of workplace: goa	
that are	Invalid	Location out of range of the application	
specied in	Location.		
the	Please	Enter expected location of workplace:	
program	enter		
	again		
Cities that	Location		Successful
are	is stored	Enter expected location of workplace: chennai	
specified			
in our		Enter Expected Job Type:	
program		1. Full Time 2. Part Time	

NOTE: The first character of the words is automatically converted into uppercase characters and the rest into lowercase characters

# 8.3.19. Job Type

Input	Expected	Actual Output	Validatio
Any form of string	Output Warning Prompt: Invalid Input.	Enter Expected Job Type:  1. Full Time 2. Part Time  Enter 1 or 2: sk Invalid Input. Job Type cannot contain an alphabet or sp	Successful
Intege r other than 1 or 2	1	Enter Expected Job Type:  1. Full Time 2. Part Time  Enter 1 or 2: 6 Invalid Input. Please fill with 1 or 2  Enter 1 or 2:	Successful
1 or 2	Respectiv e Job Type is stored	Enter 1 or 2: 1  Enter Company Type: 1. Startup 2. Small Business	Successful

# 8.3.20. Company Type

Input	Expected Output	Actual Output	Validation
Any form of string	Warning Prompt: Invalid Input.	Enter 1, 2 or 3: kjkd Invalid Input. Company Type cannot contain an alphabet o Enter 1, 2 or 3:	Successful

Integer other than 1,2 or 3	Invalid Input. Please enter 1,2 or 3	Enter 1, 2 or 3: 6 Invalid Input. Please fill with 1,2 or 3 Enter 1, 2 or 3: ■	Successful
1,2 or 3	Respective Company Type is stored	Enter 1, 2 or 3: 2 Enter Shift (D-Day/N-Night):	Successful

### 8.3.21. Shift

Input	Expected	Actual Output	Validation
	Output		
Other than D and N	Warning Prompt: Invalid	Enter Shift (D-Day/N-Night): c Invalid Input. Please fill with D or N	Successful
and in	Input.	Enter Shift (D-Day/N-Night): ■	
D or N	Respective Shift is	Enter Shift (D-Day/N-Night): d	Successful
	stored	Do you prefer a Work From Home job (Y/N): ■	

### 8.3.22. Work From Home

Inpu t	Expected Output	Actual Output	Validatio n
Othe r than Y and N	Warning Prompt: Invalid Input.	Do you prefer a Work From Home job (Y/N): c Invalid Input. Please fill with Y or N  Do you prefer a Work From Home job (Y/N):	Successfu 1
Y or N	Respectiv e Work from	Do you prefer a Work From Home job (Y/N): n	Successfu 1
	Home is stored	Prasanth  SlNo Constraints Values  1. Salary 10000  2. Location Chennai  3. Job Type Full Time  4. Company Type Corporation  5. Shift D  6. Work From Home N  Enter your no of Compulsory Constraints (Max limit is 3)	

# 8.3.23. No Of Compulsory Constraints

Input	Expected Output	Actual Output	Validatio n
Any form of string	Warning Prompt: Invalid Input.	Enter your no of Compulsory Constraints (Max limit is 3): kdjb Please enter a number between 0 - 3  Enter your no of Compulsory Constraints (Max limit is 3):	Successfu 1

Integer less than 0 or More than MAX COMP CONS	Invalid Input.	Enter your no of Compulsory Constraints (Max limit is 3): 5 Please enter a number between 0 - 3  Enter your no of Compulsory Constraints (Max limit is 3): -2 Please enter a number between 0 - 3  Enter your no of Compulsory Constraints (Max limit is 3):	Successfu 1
Integer betwee n 0 - MAX COMP CONS T	No of Compulsor y Constraints is stored	Prasanth  SlNo Constraints Values  1. Salary 10000  2. Location Chennai  3. Job Type Full Time  4. Company Type Corporation  5. Shift D  6. Work From Home N  Enter your no of Compulsory Constraints (Max limit is 3): 2  Enter the Sl. No of your Compulsory Contraint(1 - 6):	Successfu 1

# 8.3.24. Compulsory Constraint

Input	Expected	Actual Output	Validatio
	Output		n
Any form of string	Warning Prompt: Invalid Input.	Enter the Sl. No of your Compulsory Contraint(1 - 6): jdjd Input number out of range. Please enter a number between 1 - Enter the Sl. No of your Compulsory Contraint(1 - 6):	Successful

Integer less than 0 or More than NO OF CONS	Invalid Input.	Enter the Sl. No of your Compulsory Contraint(1 - 6): 8 Input number out of range. Please enter a number between 1 - Enter the Sl. No of your Compulsory Contraint(1 - 6): 0 Input number out of range. Please enter a number between 1 - Enter the Sl. No of your Compulsory Contraint(1 - 6):	Successful
Integer betwee n 0 - NO OF CONS T	Jobs are recommend ed	Enter the Sl. No of your Compulsory Contraint(1 - 6): 1 Enter the Sl. No of your Compulsory Contraint(1 - 6): 4  Salary: 4 Location: 5 Shift: 6 Job Type: 3 Company Type: 2 Nork From Home: 1  O	22 16 15
		5 Ford Matchman 30000 Chennai 82. 6 Google Coder 700000 Bangalore 76. 7 Apple Sales Exceutive 225000 New Delhi 74. 8 Apple Software Engineer 900000 Bangalore 65. 9 Adidas Designer 650000 Kolkata 64. 10 Apple Clerk 90000 Delhi 62. 11 Apollo Hospital Surgeon 750000 Hydrabad 57. 12 Apollo Hospital Doctor 850000 Mumbai 57. 13 Google Watchman 35000 Bangalore 52.	55 75 85 86 86 88 88

### 8.4. LOGIN SEEKER

# 8.4.1. Login Menu

Input	Expec ted Outp ut	Actual Output	Valida tion
Non- existe nt Seeke r	Warni ng Prom pt: Seeke r Doesn 't Exist	Enter -1 to go back Enter username: ljdsn  The given Seeker does not exist. Try again with a differenet seeker name Enter -1 to go back Enter username: ■	Succes
Existi ng Seeke r and Wron g Passw ord	Invali d Passw ord	Enter -1 to go back Enter username: Prasanth Enter Password: kshdff Incorrect Password  Enter -1 to go back Enter Password:	Succes

ng r profil r and e is Corre ct Passw ord  Cond  Cond  Constraints  Salary  Location  Job Type  Age Phone No 1234567890  1234567890  1234567890  1234567890  1234567890  1234567890  1245 Prasanna  125 prasanna  124567890  125 prasanna  125 prasanna  125 prasanna  125 prasanna  125 prasanna  124 prasanna  125 prasanna  125 prasanna  125 prasanna  125 prasanna  124 prasanna  125 prasanna  125 prasanna  124 prasanna  125 prasanna  125 prasanna  124 prasanna  125 prasanna  125 prasanna  126 prasanna  127 prasanna  127 prasanna  128 p	Existi Seeke			Succes
Company Type  Shift  Work From Home  Do you want to  => 1.Check Recommended Jobs => 2.Check Appliead Jobs  Enter -1 to logout Enter 1,2 or -1:	ng r Seeke profil r and e is Corre shown ct Passw	Phone No Email Id Qualification 10th Percentage 12th Percentage Experience College CGPA  Constraints Salary Location Job Type Company Type Shift Work From Home  Do you want to => 1.Check Recom => 2.Check Appli  Enter -1 to logout	: 1234567890 : prasanna : BE : 78.70 : 78.90 : 5 years : Ssn College Of Engine : 8.90  Values 10000 Chennai Full Time Corporation D N	sful

# 8.5. SEEKER PROFILE MENU

Inp	Expected	Actual Output	Validati
ut	Output		on
1	Jobs are recommend ed	Sl.No Company Job Salary Location  Recommended Coder 1000000 Chennai  Ford Mechanic 1000000 Chennai  Apollo Hospital Nurse 1100000 Chennai  Ford Watchman 30000 Chennai  Ford Watchman 30000 Chennai  Ford Watchman 30000 Chennai  Ford Watchman 30000 Chennai  Apple Sales Exceutive 225000 Bangalore  Apple Software Engineer 9000000 Bangalore  Apple Software Engineer 900000 Bangalore  Apple Software Engineer 900000 Bangalore  Mikata 10 Apple Clerk 90000 Delhi  11 Appllo Hospital Surgeon 750000 Hyderabad  12 Apollo Hospital Doctor 850000 Mumbai  13 Google Watchman 35000 Bangalore  Recompany  Recompany  Location  Location  Remai  Recompany  Location  Loca	Successful
2	Applied Jobs list is shown	Sl.No Company Name Job Applied Status 1 Royal Enfield coder Not Che	Successful
-1	Moves to Seeker Menu	Do you want to  => 1.Create a new Seeker account => 2.Login  Enter -1 to go back Enter 1, 2 or -1:	Successful

## 8.6. RECOMMENDATION MENU

Input	Expecte d Output	Actual Output	Validatio n
Any form of string	Invalid	5 Honda Mechanic 6 Honda Accountant  Enter -1 to go back Enter the Sl.No of the Job you want to explore: ncv Invalid Input. Please Enter a number between 1 - 6  Enter -1 to go back Enter the Sl.No of the Job you want to explore:	Successful
Beyond the range provide d	Invalid Input	Enter -1 to go back Enter the Sl.No of the Job you want to explore: 8 Invalid Input. Please Enter a number between 1 - 6 Enter -1 to go back Enter the Sl.No of the Job you want to explore:	Successful

Applies Within Successf Enter -1 to go back Enter the Sl.No of the Job you want to explore: 1 the for the ul job limit given -----Project Manager-----Salary 850000 : Chennai Location : Full Time Job Type : Corporation Company Type Work From Home Do you want to apply for this Job(Y/N): y Are you sure(Y/N): y Press any button to go back...

# 8.7. CREATE COMPANY AND JOB \*Same as CREATE SEEKER

#### 8.8. LOGIN COMPANY

\*Same as LOGIN SEEKER

## 8.9. COMPANY MENU

Inpu	Expecte	Actual Output	Validatio
t	d		n
1	Output Displays		Successfu
1	the jobs of the company	=> 1.View Jobs => 2.Add Jobs => 3.View Applicants	l
	1 3	Enter -1 to go back Enter 1,2,3 or -1: 1	
		Sl. No Job 1 Coder 2 Project Manager 3 Watchman	
		Enter -1 to logout Enter the Sl. No of Job to view its Profile:	
2	Create job	Google-	Successfu 1
	module is called	<pre>=&gt; 1.View Jobs =&gt; 2.Add Jobs =&gt; 3.View Applicants</pre>	
		Enter -1 to go back Enter 1,2,3 or -1: 2	
		Enter Job:	



#### 9. LIMITATIONS

- Scalability of the Locations and Qualifications is very low
- Cannot check if the phone number given exists or not
- Similarly, cannot check if the given Gmail is valid or not
- Cannot correctly compare Qualifications for matching
   Eg: Talent acq. Team's compulsory constraint is 10<sup>th</sup> grade for qualification, and since we compare only string literals a seeker whose qualification is BE will not satisfy while computing the recommendation percentage
- No option of forget password
- No option for editing the already existing data

#### 10. OBSERVATIONS

#### 10.1. Social Perspectives

preferences and the app gives a lot of Inclusivity:

importance to the preference of the user by

The flow of the app is solely based on the user

getting inputs

Our app does not use any complex language that the user might not be able to understand. Simplicity:

Everything is in terms of simple English so that

the user will have easy access to the app

#### 10.2. Legal Perspective

The passwords inputted by the user are hashed Hashed Password: using SHA256 hashing algorithm to increase

the security aspect of our app

Document: Verification The app demands documents including AADHAR, 10th certificate, etc to the verify the trueness of the information provided by the user. These documents will be manually

verified

### 10.3. Environmental Perspective

Reduced Code:

We have used modularity techniques and made our code efficient and effective thereby having more functionality for less code thus reducing the energy required which is evident when the number of users increase on a vast number

#### 11. LEARNING OUTCOMES

- Learnt File Handling in detail
- How to integrate Machine Learning in C
- Multi file compilation in C
- Modularity in C
- Make files
- Analysis and design reporting before implementing a project
- Problem solving and algorithm designing
- Better knowledge in C overall

#### 12. REFERENCES

• For Flow Charts

https://www.lucidchart.com/pages/examples/flowchart-maker

• Machine Learning Algorithm (KNN)

https://devcoons.com/k-nearest-neighbours-algorithm-in-c/

• Encryption of Password (SHA-256)

https://github.com/B-Con/crypto-algorithms/blob/master/sha256.c

• What is KNN

https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4