**Intelligent Agent based Job Search System**

**1.1 Overview**

Finding jobs that best suits the interests and skill set is quite a challenging task for the job seekers. The difficulties arise from not having proper knowledge on the organisation’s objective, their work culture and current job openings. Summer jobs are becoming year round side work.Even I’m rolling up my sleeves on the path of researching the best apps that will put to work on tasks, jobs and chores in one’s extra time. We set the hours and the amount of time you want to carve out for this side work.

**1.2 Applications**

In general, employers do not register themselves with these mediums to provide full details of the job specifications but instead post important details on their own website only. Also with the growing number of online job search engines, making it almost impossible for job seekers to get an overview of all relevant positions [0]. Therefore we do not always get to know all the vacancies, the nature and status of the employer to decide if this is the sort of job that is being sought for. Also at times we get flattered by the job providers profile but don’t get information about the rating of the company by the existing or past employee in terms of salary and so. Taking all these into consideration we propose to develop an intelligent agent (instead of a human agent) to perform the same search operations by interacting with the employer and job search coordinator agents. We propose to use an agent based utility concept to provide suitability profiling based on configurable factors such as distance from work, days and shift requirements, work environment, safety and hazard considerations, remuneration, skill-set, etc

**1.3PROBLEM DEFINITION:**

Our main aim is to provide an easy going application for people who are in search for small paid work in our local area. It is for users who are in need of some quick cash and willing to do small works.

**1.4 AIM OF THE PROJECT**

To understand the problems and struggle faced by the rural people in their daily life and try to relate the solution to their problems by applying the basic understanding of our engineering knowledge.

**Literature Survey:**

In this paper they introduced a prototype job portal which uses semantically annotated job offers and applicants. In their opinion, using Semantic Web technologies substantially increase market transparency, lower transaction costs and speed up the procurement process. How-ever adding semantics is not a panacea for everything. We identify some outstanding problems in job search using the system and outline how the technique of query approximation can be the basis for a solution. Through an Industry-Research co-operation we are extending the prototype with these semantic techniques to demonstrate a more accurate job search. [0]

The advent of software agents gave rise too much discussion of just what such an agent is, and of how they differ from programs in general. Here we propose a formal definition of an autonomous agent which clearly distinguishes a software agent from just any program. We also offer the beginnings of a natural kinds taxonomy of autonomous agents, and discuss possibilities for further classification. Finally, we discuss subagents and multi-agent systems.

Intelligent agents are a new paradigm for developing software applications. More than this, agent-based computing has been hailed as ‘the next significant breakthrough in software development ’(Sargent, 1992), and ‘the new revolution in software ’(Ovum, 1994). Currently, agents are the focus of intense interest on the part of many subfields of computer science and artificial intelligence. Agents are being used in an increasingly wide variety of applications, ranging from comparatively small systems such as email filters to large, open, complex, mission critical systems such as air traffic control. At first sight, it may appear that such extremely different types of system can have little in common. And yet this is not the case: in both, the key abstraction used is that of an agent our aim in this article is to help the reader to understand why agent technology is seen as a fundamentally important new tool for building such a wide array of systems. More precisely, our aims are five-fold:

• to introduce the reader to the concept of an agent and agent-based systems.

• to help the reader to recognize the domain characteristics that indicate the appropriateness of an agent-based solution.

• to introduce the main application areas in which agent technology has been successfully deployed to date.

• to identify the main obstacles that lie in the way of the agent system developer.

Agents and agent systems are becoming more and more important in the development of a variety of fields such as ubiquitous computing, ambient intelligence, autonomous computing, intelligent systems and intelligent robotics. The need for improvement of our basic knowledge on agents is very essential. We take a systematic approach and present extended classification of artificial agents which can be useful for understanding of what artificial agents are and what they can be in the future. The aim of this classification is to give us insights in what kind of agents can be created and what type of problems demand a specific kind of agents for their solution.

Functional and non functional requirements: Based on the Job Search System application’s requirements, I have made the choice of frameworks to be used. This application primarily consists of an android application that displays the jobs that are available and applied by the applicants. This android application is developed by using the Native Android Framework: Used for building an android application, Google Maps SDK: Used for integrating the application with Google maps, Firebase: Used for the mobile clients without any extensive backend programming.

**3. SYSTEM ANALYSIS**

**EXISTING SYSTEM**

The existing system for job recruitment includes traditional methods like Employment agencies, advertising through newspapers, televisions and radios, college fairs etc., which are too slow and stressful. With the advancement of internet, jobseekers rely on the online job portals, which makes the job search efficient. Again, most of these are limited to the web/desktop applications, which requires jobseekers to have a laptop or desktop connected to internet and is not handy. And disadvantages include: Time Consuming, Stressful, Challenging.

**PROPOSED SYSTEM**

Job Search System is a Java-based android application that provides functionalities of erecruitment on portable devices like Android based smart phones/tablets. The applications do not require internet to perform the desired functionalities. Advantages: Cost and Time efficient, Portable

Job Search System is developed to provide an effective means for the employers to post job openings with required qualification to have a better penetration into the job market and jobseekers to find out the information regarding the current openings in the organization or in the market. In addition, job seekers can view the reviews provided by the applicants to make necessary improvements in their system if needed. Job search System is an android application providing flexibility for the users.

**SYSTEM REQUIREMENTS SEPEFICTATION**

**4.1 FUNCTIONAL REQUIREMENTS**

In [software engineering](https://en.wikipedia.org/wiki/Software_engineering) and [systems engineering](https://en.wikipedia.org/wiki/Systems_engineering), a **functional requirement** defines a function of a [system](https://en.wikipedia.org/wiki/System) or its component, where a function is described as a specification of behavior between outputs and inputs.

Functional requirements may involve calculations, technical details, data manipulation and processing, and other specific functionality that define what a system is supposed to accomplish. Behavioral requirements describe all the cases where the system uses the functional requirements, these are captured in [use cases](https://en.wikipedia.org/wiki/Use_case). Functional requirements are supported by [non-functional requirements](https://en.wikipedia.org/wiki/Non-functional_requirement) (also known as "quality requirements"), which impose constraints on the design or implementation (such as performance requirements, security, or reliability). Generally, functional requirements are expressed in the form "system must do <requirement>," while non-functional requirements take the form "system shall be <requirement>." The plan for implementing functional requirements is detailed in the system design, whereas *non-functional* requirements are detailed in the system architecture.[[4]](https://en.wikipedia.org/wiki/Functional_requirement#cite_note-AdamsNon15-4)[[5]](https://en.wikipedia.org/wiki/Functional_requirement#cite_note-J%C3%B6nssonImpact06-5)

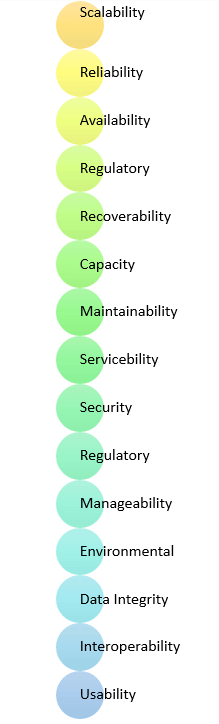
As defined in [requirements engineering](https://en.wikipedia.org/wiki/Requirements_analysis), functional requirements specify particular results of a system. This should be contrasted with non-functional requirements, which specify overall characteristics such as cost and [reliability](https://en.wikipedia.org/wiki/Reliability_engineering). Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.[[4]](https://en.wikipedia.org/wiki/Functional_requirement#cite_note-AdamsNon15-4)

In some cases a requirements analyst generates use cases after gathering and validating a set of functional requirements. The hierarchy of functional requirements collection and change, broadly speaking, is: user/[stakeholder](https://en.wikipedia.org/wiki/Project_stakeholder) request → analyze → use case → incorporate. Stakeholders make a request; systems engineers attempt to discuss, observe, and understand the aspects of the requirement; use cases, entity relationship diagrams, and other models are built to validate the requirement; and, if documented and approved, the requirement is implemented/incorporated.[[6]](https://en.wikipedia.org/wiki/Functional_requirement#cite_note-MITRESys14-6) Each use case illustrates behavioral scenarios through one or more functional requirements. Often, though, an analyst will begin by eliciting a set of use cases, from which the analyst can derive the functional requirements that must be implemented to allow a user to perform each use case.

**4.2 NON-FUNCTIONAL REQUIREMENTS**

**NON-FUNCTIONAL REQUIREMENT** (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, *“how fast does the website load?”* Failing to meet non-functional requirements can result in systems that fail to satisfy user needs.

Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are> 10000. Description of non-functional requirements is just as critical as a functional requirement.



* Usability requirement
* Serviceability requirement
* Manageability requirement
* Recoverability requirement
* Security requirement
* Data Integrity requirement
* Capacity requirement
* Availability requirement
* Scalability requirement
* Interoperability requirement
* Reliability requirement
* Maintainability requirement
* Regulatory requirement
* Environmental requirement

**Advantages of Non-Functional Requirement**

Benefits/pros of Non-functional testing are:

* The nonfunctional requirements ensure the software system follow legal and compliance rules.
* They ensure the reliability, availability, and performance of the software system
* They ensure good user experience and ease of operating the software.
* They help in formulating security policy of the software system.

**Disadvantages of Non-functional requirement**

Cons/drawbacks of Non-function requirement are:

* None functional requirement may affect the various high-level software subsystem
* They require special consideration during the software architecture/high-level design phase which increases costs.
* Their implementation does not usually map to the specific software sub-system,
* It is tough to modify non-functional once you pass the architecture phase.

**KEY LEARNING**

* A non-functional requirement defines the performance attribute of a software system.
* Types of Non-functional requirement are Scalability Capacity, Availability, Reliability, Recoverability, Data Integrity, etc.
* Example of Non Functional Requirement is Employees never allowed to update their salary information. Such attempt should be reported to the security administrator.
* Functional Requirement is a verb while Non-Functional Requirement is an attribute
* The advantage of Non-functional requirement is that it helps you to ensure good user experience and ease of operating the software
* The biggest disadvantage of Non-functional requirement is that it may affect the various high-level software subsystems.

**4.3 HARDWARE REQUIRMENTS:**

* System    :   I3.
* Hard Disk  :   40 GB.
* Ram    :   4 GB.

**4.4 SOFTWARE REQUIRMENTS:**

* Operating system   : Windows 8 .
* Coding Language  : python
* DATABASE : MYSQL
* FRAMEWORK : DJANGO

**5.SYSTEM DESIGN**

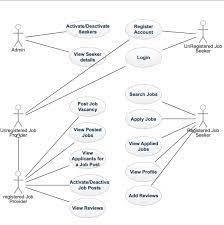
**5.1 PROPOSED METHODLOGY**

The existing system for job recruitment includes traditional methods like Employment agencies, advertising through newspapers, televisions and radios, college fairs etc., which are too slow and stressful. With the advancement of internet, jobseekers rely on the online job portals, which makes the job search efficient. Again, most of these are limited to the web/desktop applications, which requires jobseekers to have a laptop or desktop connected to internet and is not handy. And disadvantages include: Time Consuming, Stressful, Challenging

Proposed System: Job Search System is a Java-based android application that provides functionalities of erecruitment on portable devices like Android based smart phones/tablets. The applications do not require internet to perform the desired functionalities. Advantages: Cost and Time efficient, Portable

Purpose of the System: Job Search System is developed to provide an effective means for the employers to post job openings with required qualification to have a better penetration into the job market and jobseekers to find out the information regarding the current openings in the organization or in the market. In addition, job seekers can view the reviews provided by the applicants to make necessary improvements in their system if needed. Job search System is an android application providing flexibility for the users.

**5.2 SYSTEM ARCHITECTURE**



**5.3SYSTEM DESIGN**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

## GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.





# CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



COLLABRATION DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



## IMPLEMENTATION:

## MODULES:

Signup Here : link to get below signup screen

Submit : button to get below output

Login Here : link to login as company

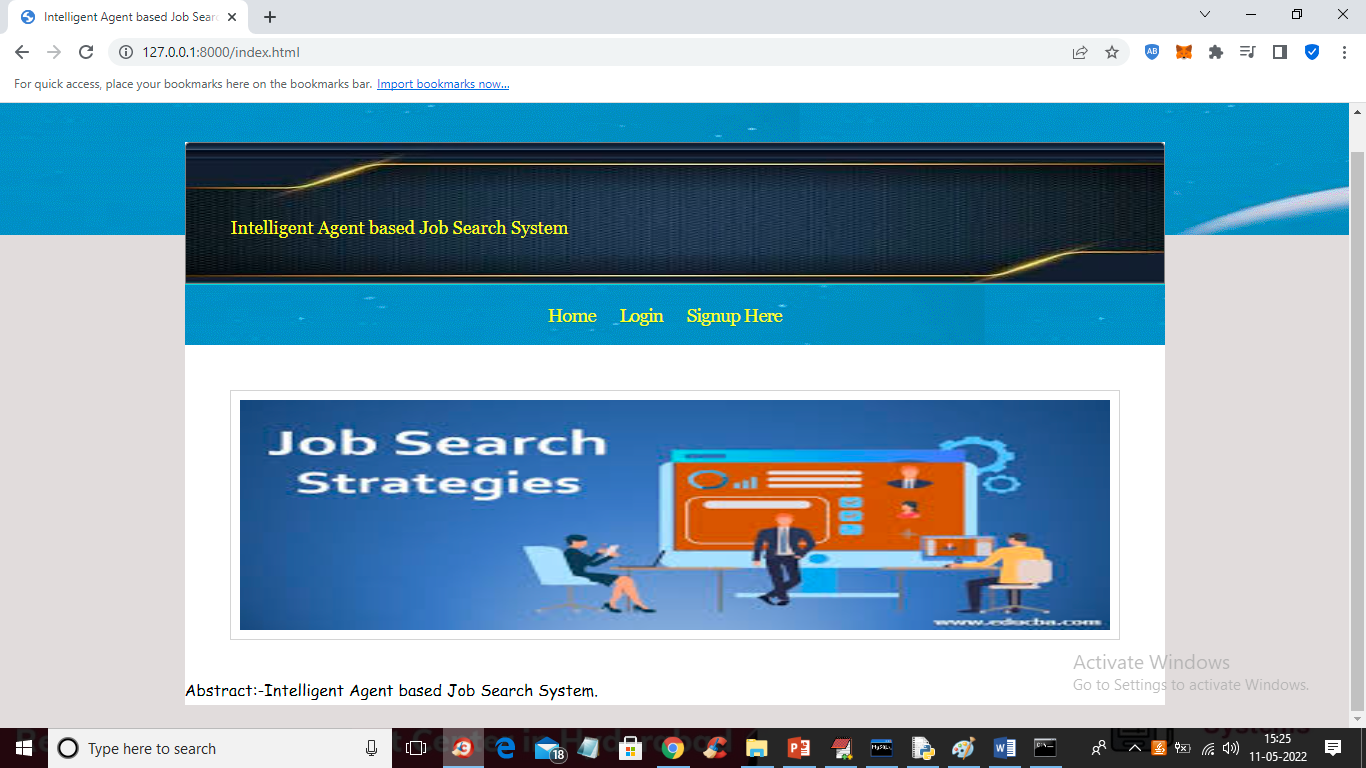
Post Jobs : Jobs’ link to post the new JOB

Activate/Deactivate Job : link to get list of jobs posted by company and then activate or deactivate

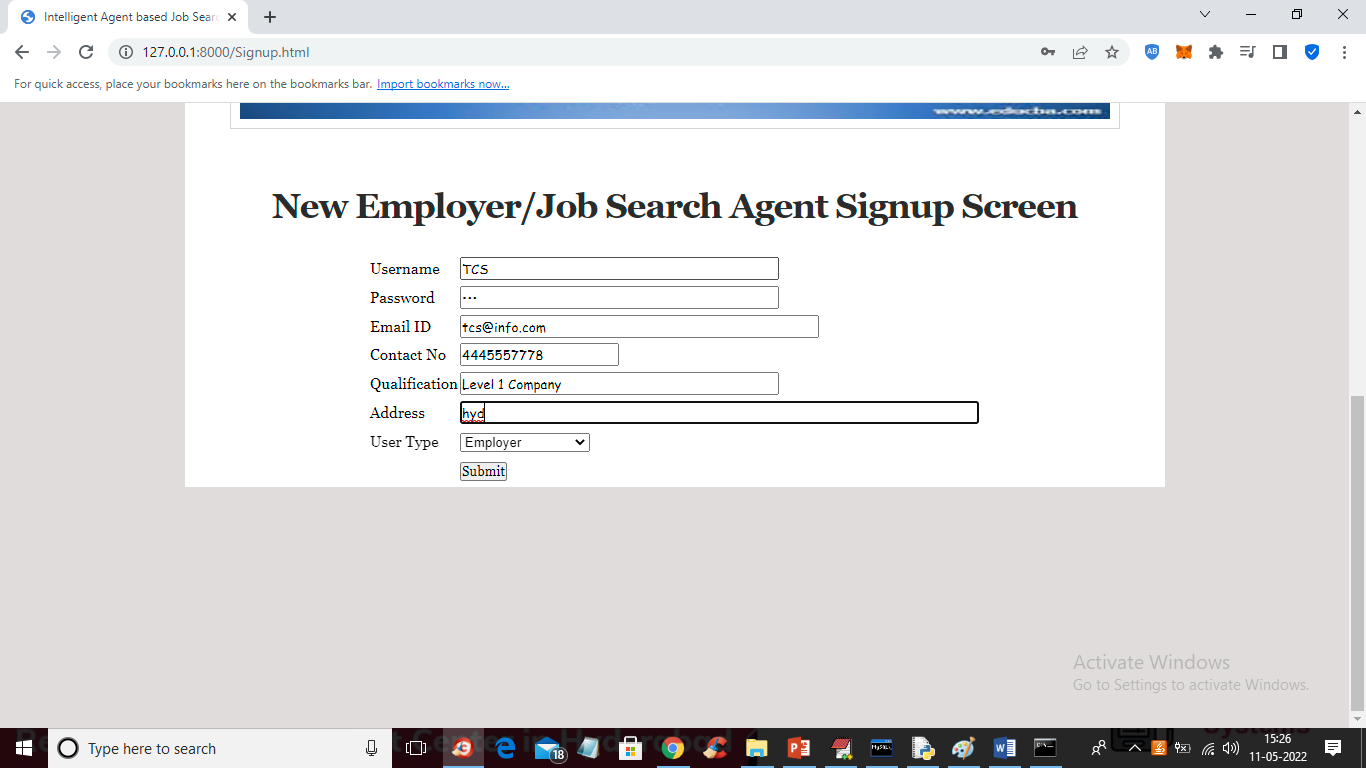
Click Here : link to either activate or deactivate JOB. Now logout and login as ‘job applicant

**6.OUTPUT SCREENS**

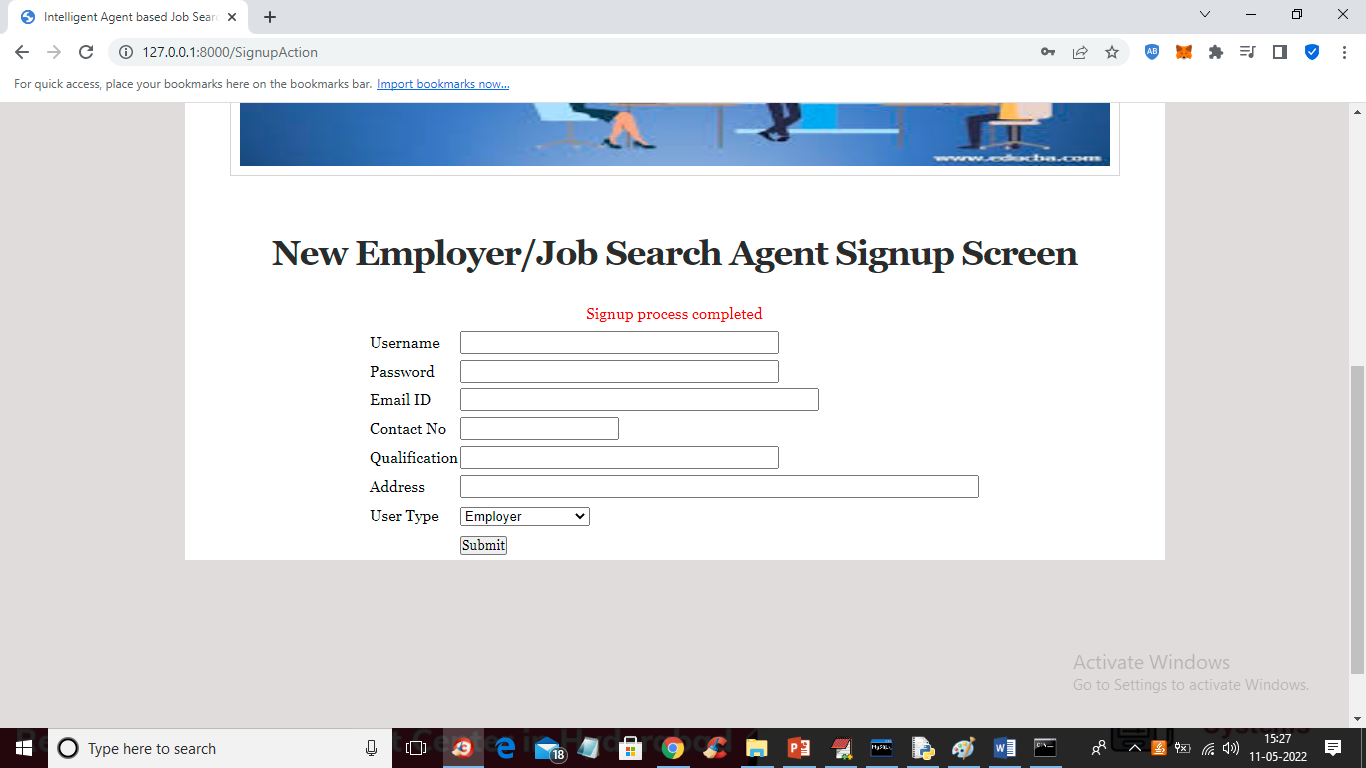
**HOME PAGE**



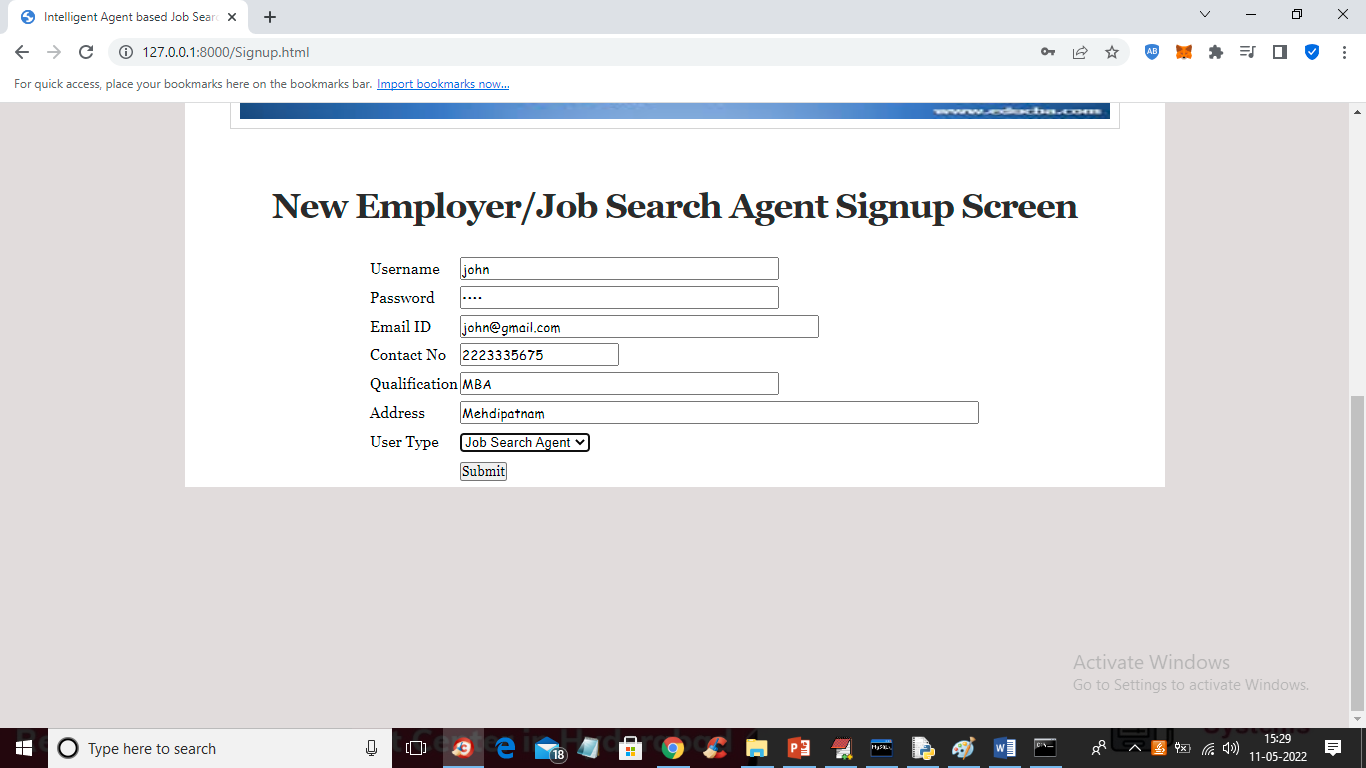
In above screen click on ‘Signup Here’ link to get below signup screen



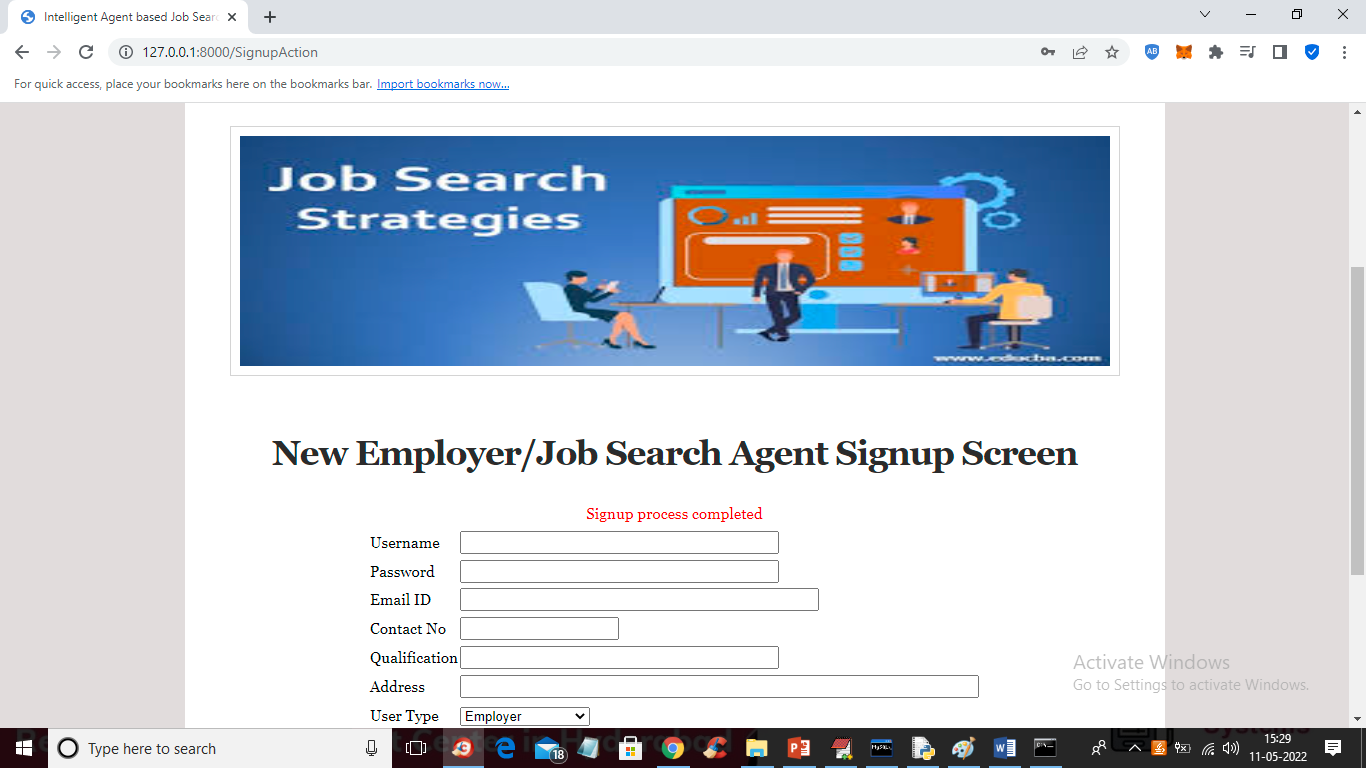
In above screen company is entering signup details and press button to get below screen



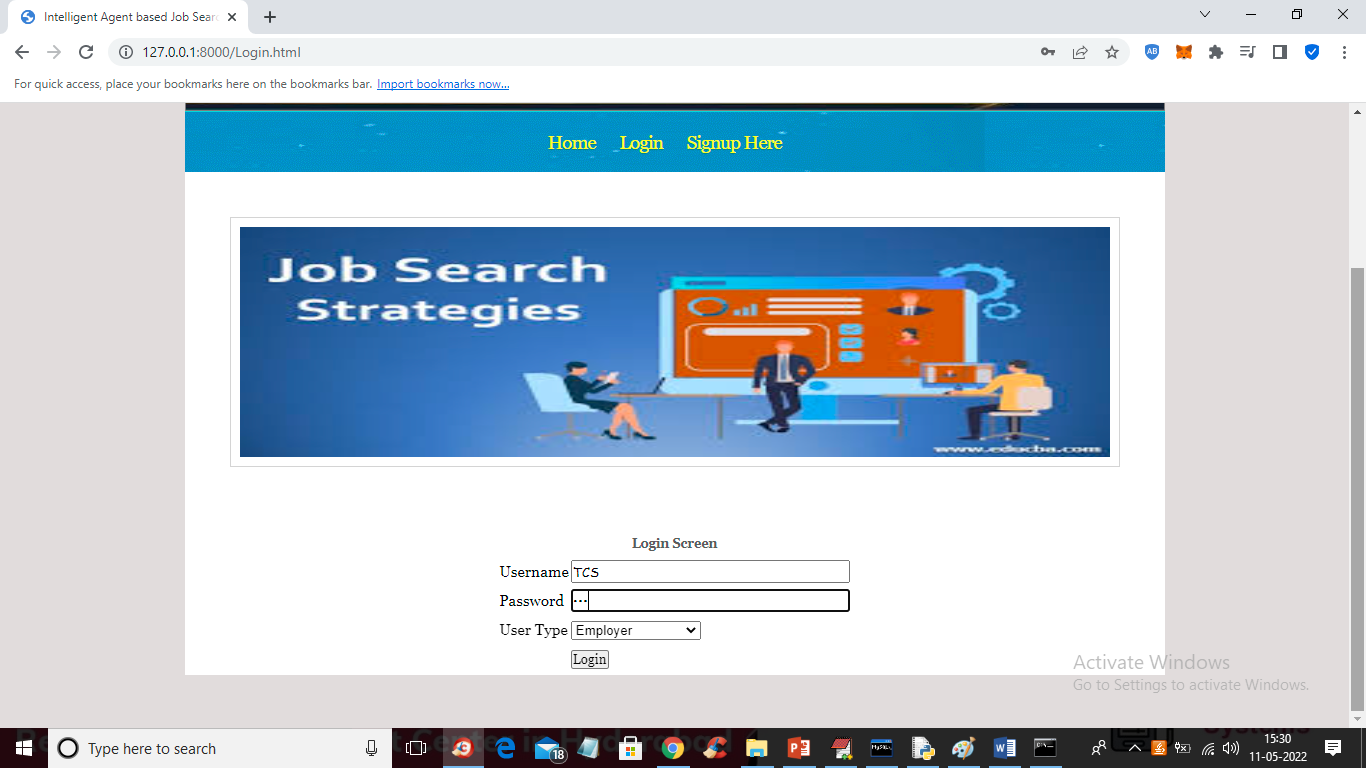
In above Screen Company signup process completed and now signup one application like below screen



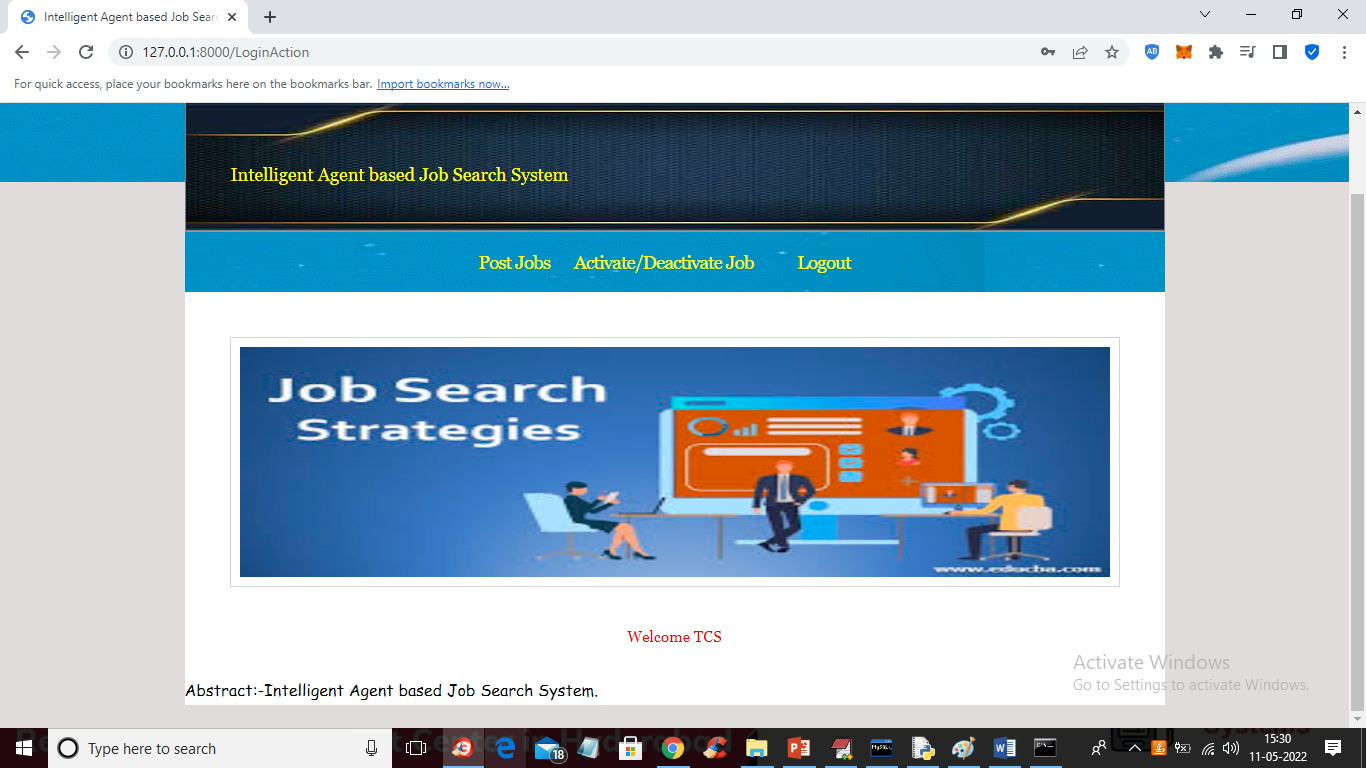
In above screen job applicant is signing up and then click on ‘Submit’ button to get below output



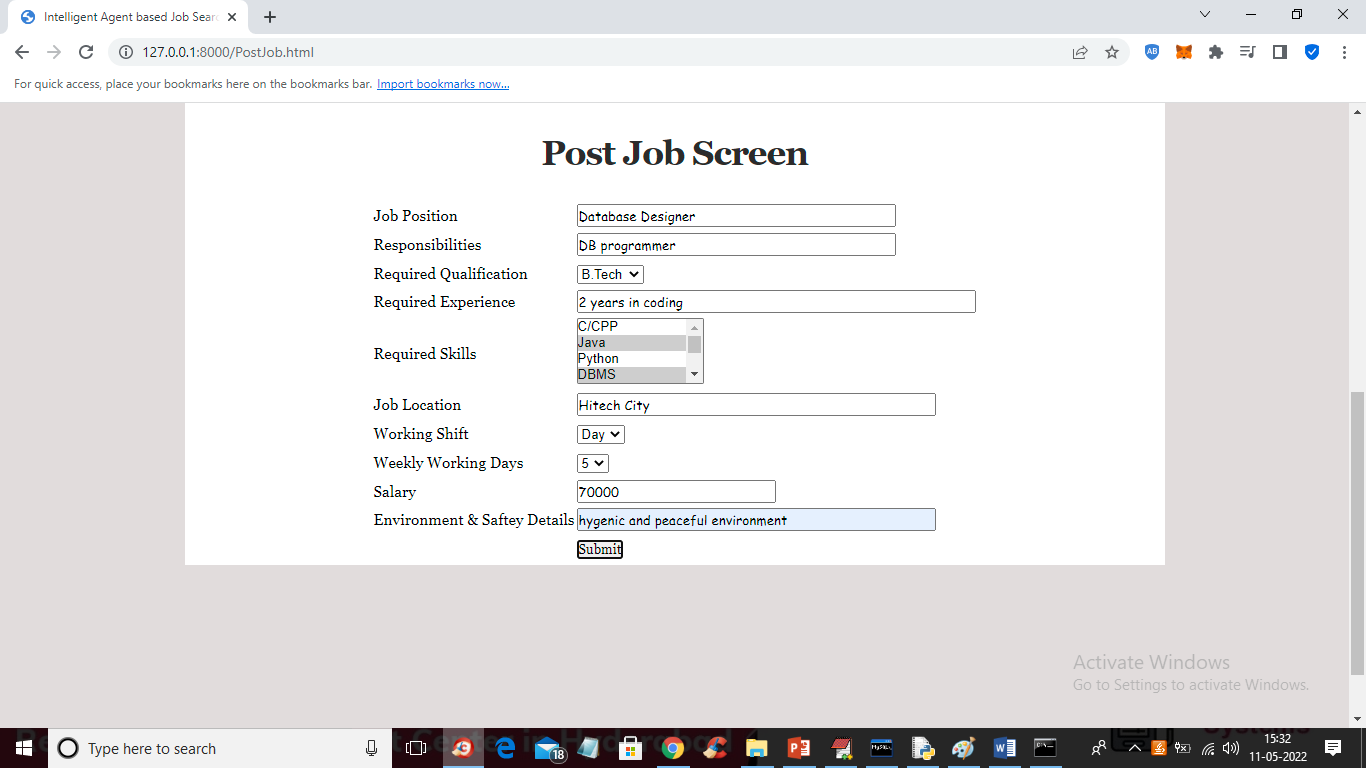
In above screen applicant signup completed and now click on ‘Login Here’ link to login as company



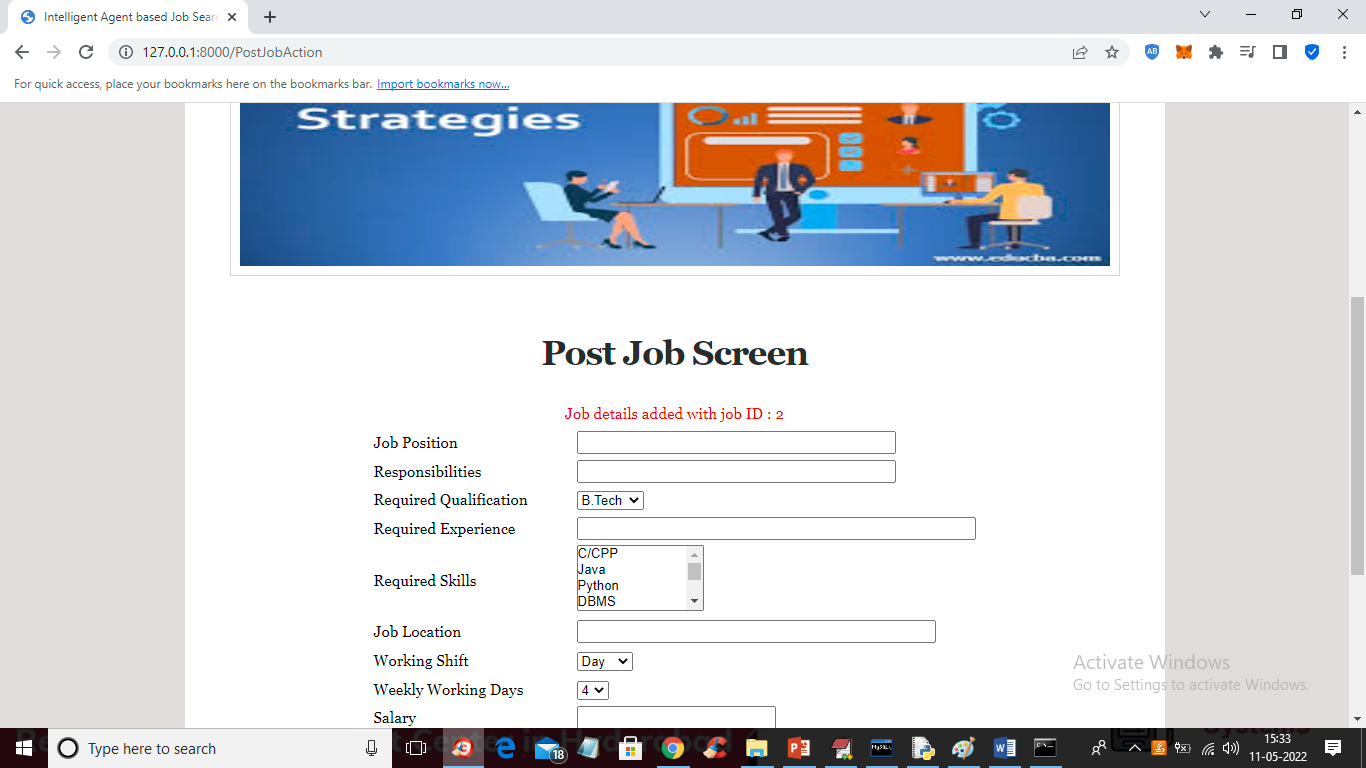
In above screen company is login and after login will get below screen



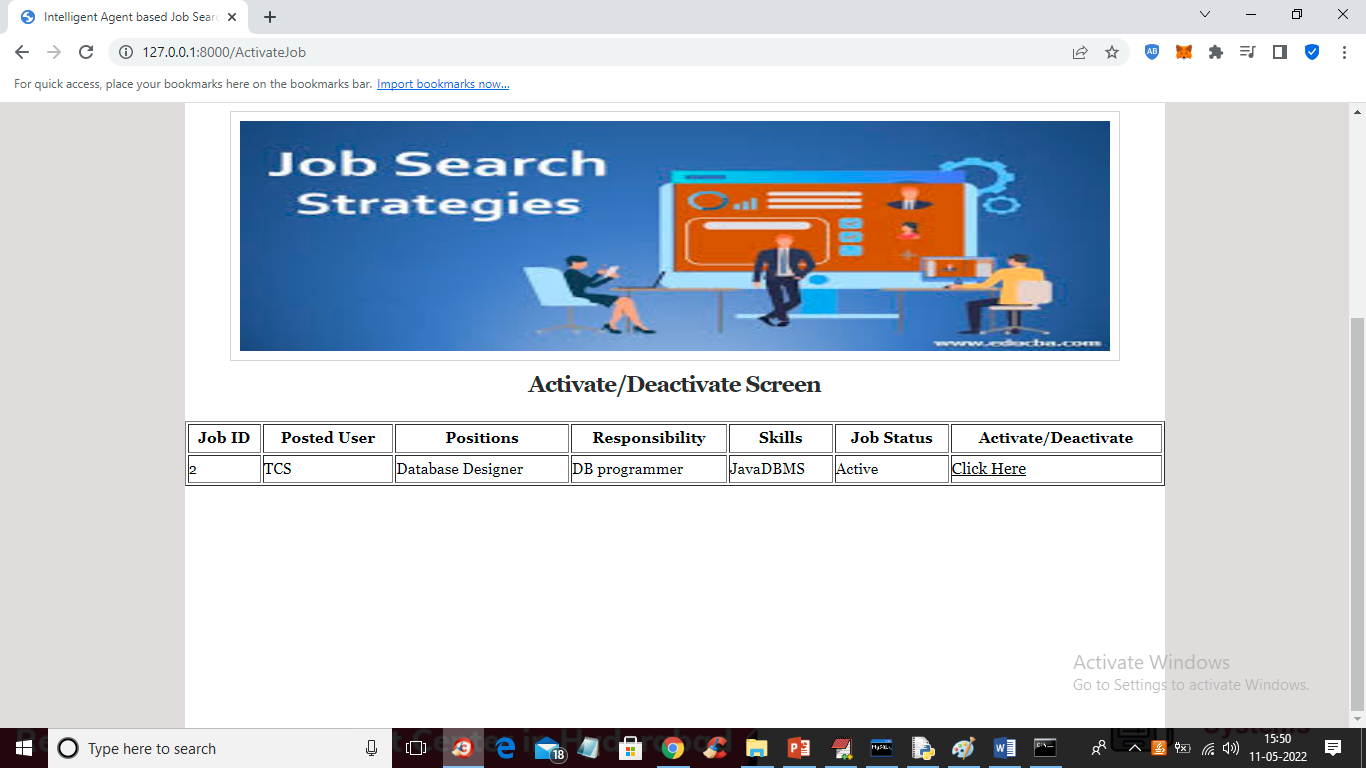
In above screen company can click on ‘Post Jobs’ link to post the new JOB



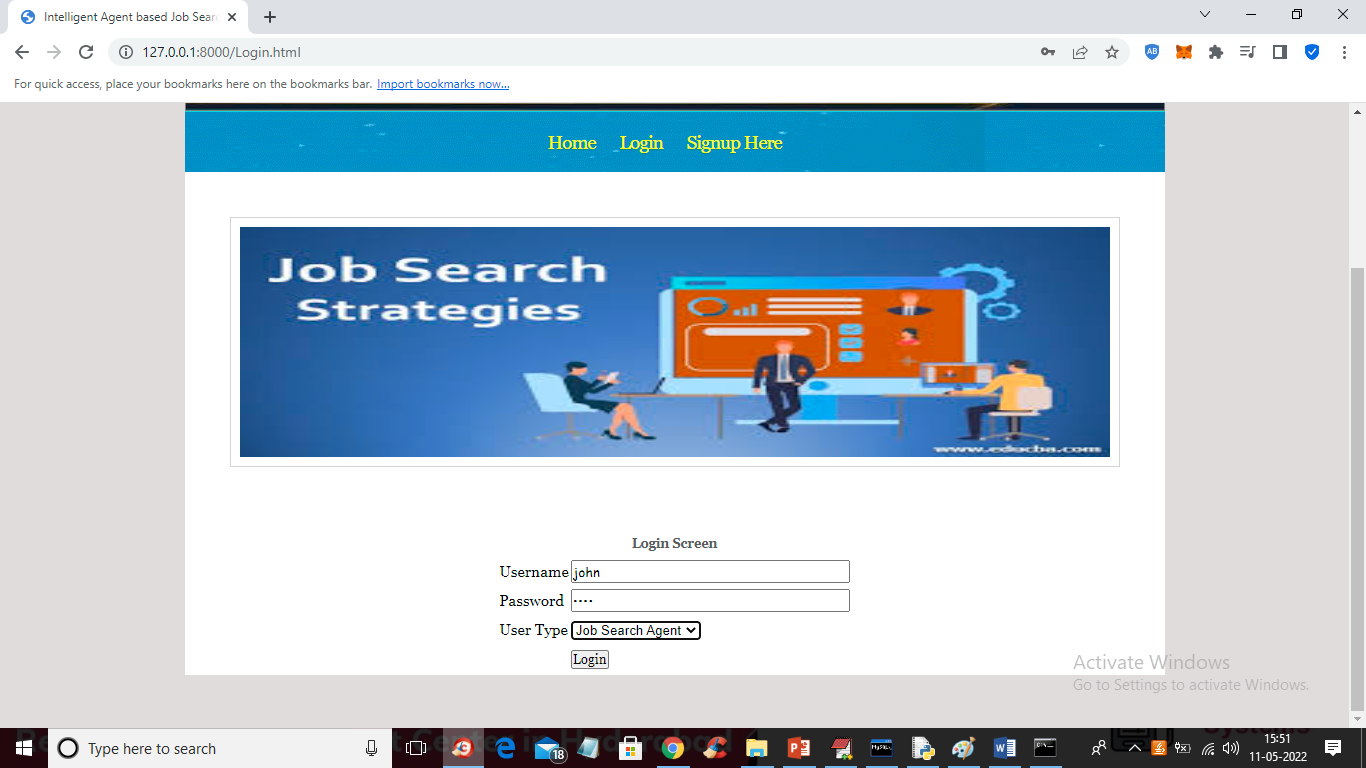
In above screen company is posting the job and now press button to get below output



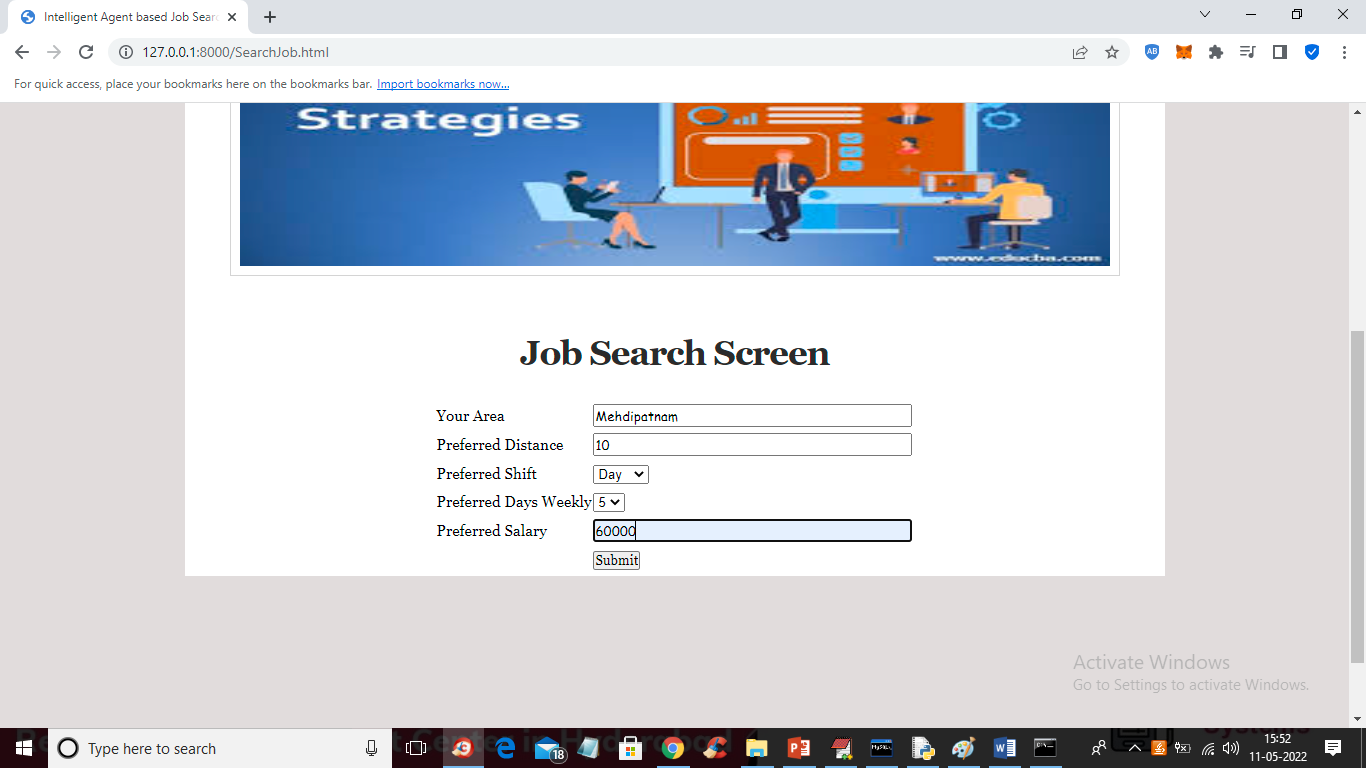
In above screen job details added and now click on ‘Activate/Deactivate Job’ link to get list of jobs posted by company and then activate or deactivate



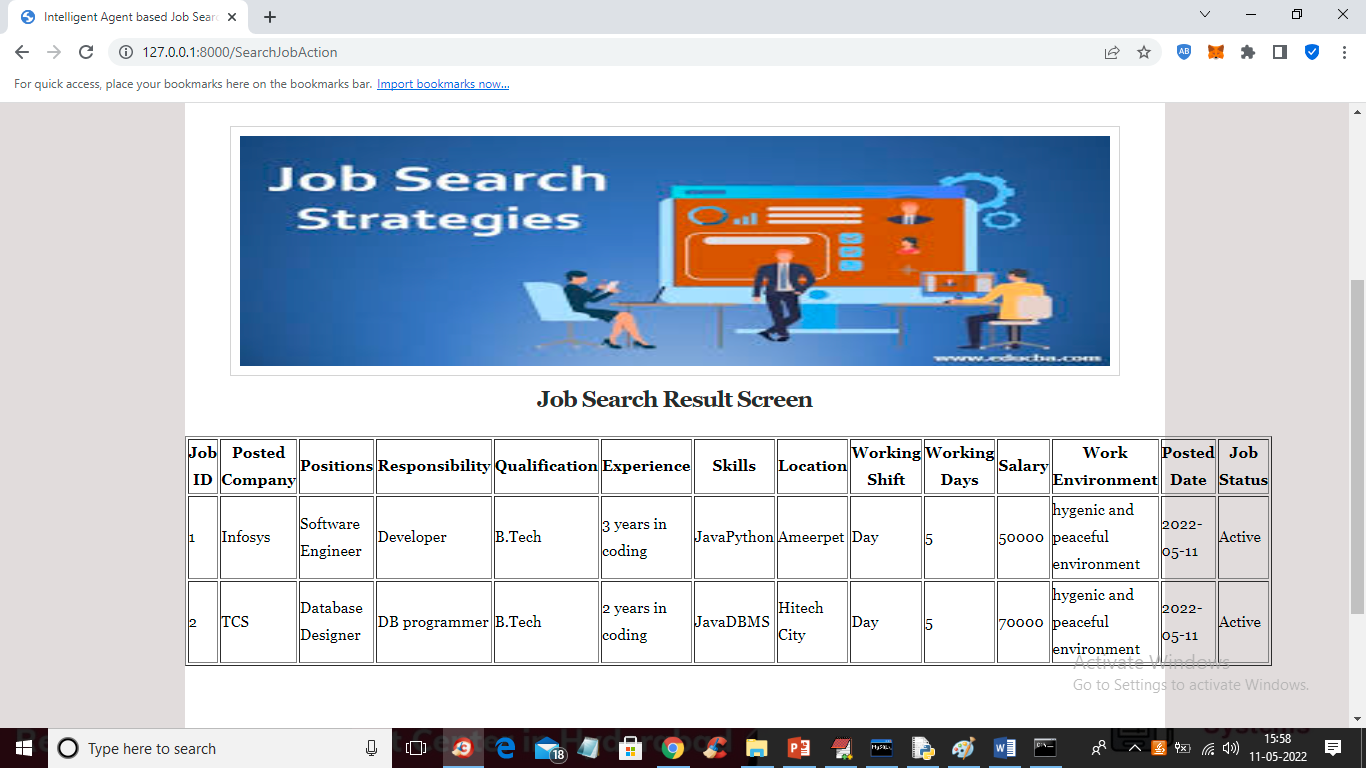
In above screen company can click on ‘Click Here’ link to either activate or deactivate JOB. Now logout and login as ‘job applicant’



In above screen job applicant is login and after login will get below screen



For above search we got below fuzzy search result



Similarly company will post JOBS and job seekers will search jobs

**7.CONCLUSION :**

Job Search is a very involved process that could require hours of interaction with different search sites, applications, human agents, etc. The developed system intelligently anticipates the needs of the user and makes intelligent decisions based on fuzzy preference rules and dynamically make location, salary markup and markdown, and allowances choices that are perceived as beneficial to the user. This is evident in the results presented in the form of scenarios and supporting screenshots. The system could be extended to include a secure application process where the applicant’s experience and education is verified possibly by including biometric data along with the job application details which has been published elsewhere. In addition the job search process could enhance the calculation of utility by including risk factors of success in choosing one job over another. This could enhance the probability of applying for the job that would be most suitable for an applicant on many levels.

This project fulfils the primary requirements of the job seekers and employers. It can be extended in several ways – We can provide recommendations and email updates for new job postings based on the job seeker’s search history. Since, the job seekers might be interested in building a strong Resume, we can provide tips and information for the same. We can also provide templates for building the Resumes which might interest most applicants. The mobile application is developed fulfilling the functionalities of job seeker, it can be extended to support functionalities of Employer as well.

**REFERENCES :**

[0]Mochol, Malgorzata, Holger Wache, and Lyndon Nixon. "Improving the accuracy of job search with semantic techniques." Berlin, Germany, 2007

[1]Franklin, Stan, and Art Graesser. " Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents." Third International Workshop on Agent Theories Architectures and Languages. Springer-Verlag, 1996.

[2]Jennings, N. R., and M. Wooldridge. Applications of Intelligent Agents. London: University of London, 1998.

[3]Hayes-Roth, B. "An Architecture for Adaptive Intelligent Systems." Artificial Intelligence: Special Issue on Agents and Interactivity, 1995: 72, 329-365.

[4] h#ps://www.genmymodel.com/use-case-diagramonline