

Jobu Online Android Framework For Job Portal Systems

Ms. Pallavi Hongekar

P. G. Scholar, Department of MCA,

KLS Gogte Institute of Technology, Udyambag, Belagavi.

Dr. Sunita S. Padmannavar

Asst. Professor, Department of Master of Computer Applications

KLS Gogte Institute of Technology, Belagavi, Karnataka, India

Abstract- Resume parsing is a technique which will automate the process of populating the information on the specific form and sends an output which will be stored into the database. This reduces the candidate's task of manually entering the information. After uploading the resume it will be parsed such that it will extract parameters from the resume and then matches these parameters with required attributes which are predefined.

Keywords – Resume Parser, JSON Data, JD Parser, API, Job search, Jobseeker

I. INTRODUCTION

In today's competitive world, the education among the people is increasing rapidly that, it is difficult for them to get job easily. It is also difficult for them to find the best suitable job that match with their skills. The companies also want the people who are best in their fields. And then it becomes difficult to find the candidates who are intelligent enough to be hired. The work for the companies also increases to find the candidates who can fulfill their requirements. Thinking about these problems, there should be some way to handle recruitment process and make the work less complex. Job portal application has come up with the facility to make this job searching process easier. Candidates uploads their resumes on these job portal applications. These resumes may be in any format and also the information that it contains may be like full sentences. At this time checking these resume and taking the required information about a candidate from it becomes difficult and also time consuming task. This problem can be solved by using the resumining parsing process into a job portal application. This feature will be able to automate task of checking resume details. Rchilli Resume Parser API are useful in this process. Managing the resume information is much easier with this API. Now there is no need to manually enter data. Now we can parse both resume and job description with resume parser and JD parser and then use this data as needed. The API works with all types of resume documents such as PDF or doc.

II. LITERATURE SURVEY

The system offers users an easy & effective means of controlling their various home appliances from a remote location i.e. without physically being present at home. The system makes use of the internet to enable remote access to the various home appliances. The web-controlled embedded system using mobile phone gives tremendous freedom to control devices during movements or away from site. This system is a economical setup which can be afforded by majority. The basic vision of the system is to provide a convenient & secure system to the user, which would aid the high degree of mobility & control, we aim to achieve now a days (Padmannavar, 2015).

Currently web applications, mobile applications with all functionality that provides the user to find the jobs and post the jobs. These applications are web apps or native apps. The existing system doesn't have parsing mechanism feature so the user manually has to fill information and also profile details separately.

Proposed System - The proposed application allows candidates to upload their resume without any limitations on its format. These resumes are then analyzed by the resume parser API and provides us JSON data. The JSON data is then used to store the required information into the database and also populate into the specific forms in an application.

III. IMPLEMENTATION

Designing an automated application which extracts candidate details from unstructured resume and then convert it into structured format. The information is extracted according to candidate's personal information, education, experience and skills.

Below diagram represents different use cases of the application. It describes how different users interact with different modules of system.

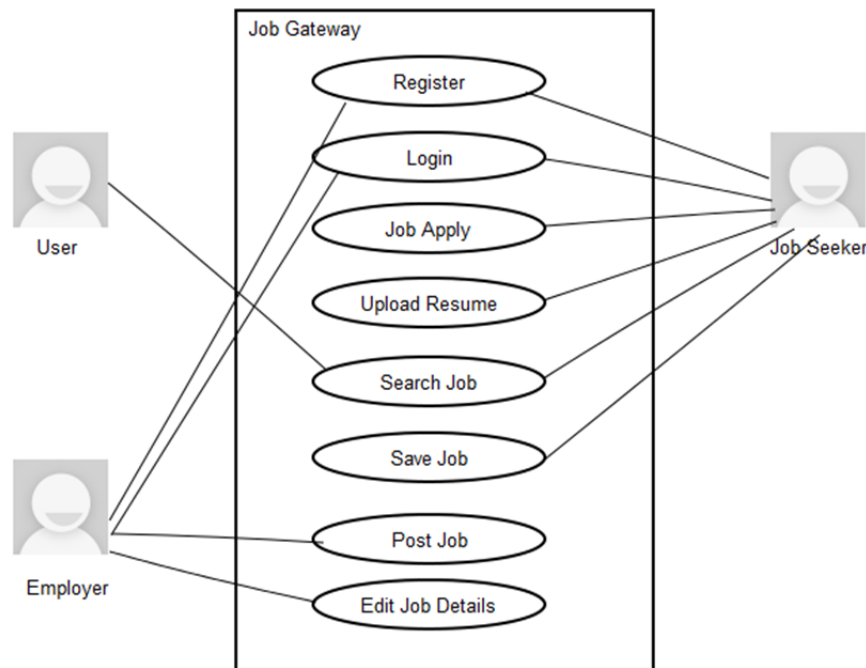


Fig.1 Use Case Diagram

System architecture consists of following modules.

A. Job Seeker module

To get started with the application the user has to register with the application and verify email while this registration process. This will confirm whether the user is valid or not. The email verification process is done by generating an OTP and then send it to the email address that was provided by the user at the time of registration. Once the OTP is verified the user is successfully registered now. Next user can login to the application by using his username and password which he/she set during registration. Uploading the user details manually is time consuming task. The proposed application allows the user to upload the resume which is a quick process as compare to manually entering the data.

B. Resume Parsing

This is an important module which will parse the resume by using a third client API called as Rchilli Resume Parser API. The API takes the resume in any format and then analyses it by matching the resume data with the already predefined keywords and then generates the structured resume in the form of JSON data. And returns this JSON data. From this JSON data we are fetching the required details and then storing them into the database.

C. Employer

Employer is responsible for registering the company into our application and then post jobs. Posting jobs with the required skills and other knowledge that is needed.

D. Match Jobs

This is a matching engine which will match the candidates information with the jobs posted. And notifies candidate about the matching job and also notifies employer about the matching candidate.

Screen shots of home screen, registration and fill out information designed modules is as shown in figure 2 below

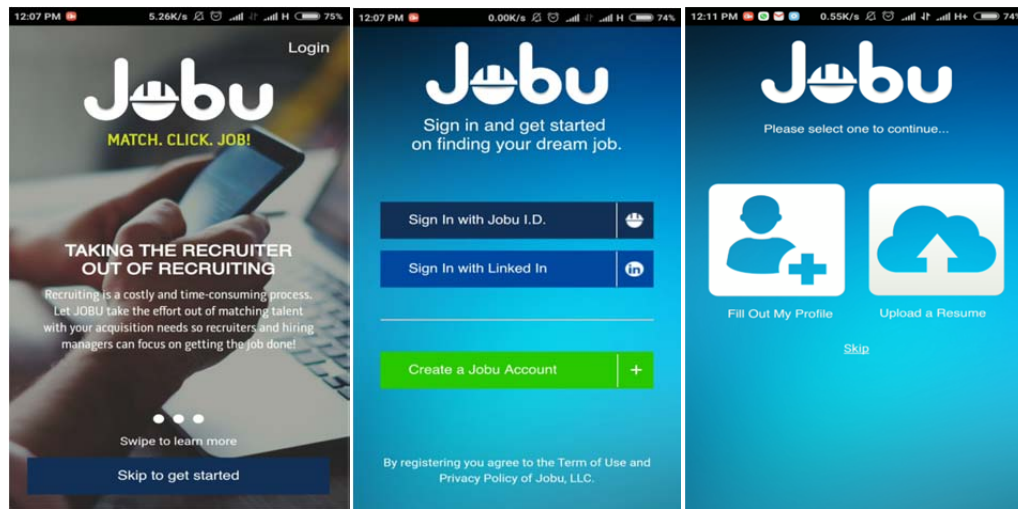


Fig. 2 Screen shots of designed modules

FUTURE ENHANCEMENTS

The current application works for all types of the person who is in need of jobs. But application will be good and user friendly when it distinguishes different type of the person who is in need of jobs such as, whether the person, who is in need of job, is a fresher or an experienced employee. The location intelligence feature can be implemented so that it will notify the person who is in need of job to get best matching job based on his/her location. The application can be connected with colleges. This will be helpful for the companies to hire the fresher who are willing to work in specific technology in which they are good at.

CONCLUSION

The resume parsing is very useful technique. It allows an easy way to manage the resumes that are uploaded by the candidates and it also reduces the work of an employer to go through those texts or essay format resumes. As resume parser fetches the important data which is required for the job and then represents it into the JSON format which will be more understandable and will be quick to process.

The current application works for all types of job seekers. But application will be good and user friendly when it distinguishes different type of Job seekers such as, whether the Job seeker is a fresher or an experienced employee. The location intelligence feature can be implemented so that it will notify Job seeker to get best matching job based on his/her location. The application can be connected with colleges. This will be helpful for the companies to hire the fresher who are willing to work in in specific technology in which they are good at.

REFERENCES

- [1] <https://rchilli.com/solutions/resumeparser-api/>
- [2] Sunita Padmannavar, "Controlling Home Appliances Using Embedded Systems", International Journal of Modern Trends in Engineering and Research ,ISSN Online : 2349-9745, ISSN Print : 2393-8161, Volume 2, Issue 6, June-2015
- [3] Vivek Kumar Sehgal ; Akshay Jagtiani ; Meha Shah ; Anupriya Sharma ; Arpit Jaiswal ; Dhananjay Mehta, "Job Portal - A Web Application for Geographically Distributed Multiple Clients", Artificial Intelligence, Modelling and Simulation (AIMS), 2013 1st International Conference on 3-5 Dec. 2013, **Publisher: IEEE**