

The Job Application Tracker

Prithvish Doshi
pdoshi@ncsu.edu
NC State University
North Carolina, USA

Seeya Prabhumi
spprabhu@ncsu.edu
NC State University
North Carolina, USA

Chao-Ting, Hung
chung4@ncsu.edu
NC State University
North Carolina, USA

Shao-Yo Chao
schao2@ncsu.edu
NC State University
North Carolina, USA

Vraj Chokshi
vschokshi@ncsu.edu
NC State University
North Carolina, USA

ABSTRACT

The idea of this document is to summarize the purpose, functionality and the overall design of the application built as a part of the SE Fall 2021 batch project. The Job Application Tracker provides a one stop solution to suggest all relevant job opportunities currently available in the market and also a direct link to the application portal. The technologies used and future scope is summarized in the following sections.

KEYWORDS

Application, Job Tracking, Opportunities, Status, System, Management Tool

ACM Reference Format:

Prithvish Doshi, Seeya Prabhumi, Chao-Ting, Hung, Shao-Yo Chao, and Vraj Chokshi. 2018. The Job Application Tracker. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Applying for 1000+ jobs and tracking each one of it everyday is definitely not a simple task to do. Moreover knowing when and where to apply is also quite a tedious job. The main aim of building this tracking system is to help adults know of all relevant job opportunities available at the time in the market space and apply for the same.

The application will provide a consolidate view of what all opportunities are available and where-all have you applied until now. It will also show a list of all positions that did not go through.

2 APPLICATION OVERVIEW

As noted in the introduction, the Application built will provide a consolidated view of the Wish-List - opportunities available, Applied - a list of all opportunities you applied to and Rejected - All the positions that did not go through.

Unpublished working draft. Not for distribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted by ACM, provided that the copies are not made for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Woodstock '18, June 03–05, 2018, Woodstock, NY

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/10.1145/1122445.1122456>

2021-09-30 00:16. Page 1 of 1–2.

How would the application work? As the user enters the interface, he will have the option to either add a particular entry of the opportunity available or enter relevant keywords he is looking out for. Based on the keywords available job opportunities would show up and the user will be given the options to "Accept/ Reject" a particular entry he wants to add to the Wish List.

The Details shown will include - Company, Position and Application deadline date. Once Applied, the entry will move to "List of Applied"

The Application would also show the list of opportunities, one couldn't get through to track the same.

2.1 Future Scope

The idea is to have multiple job application sites linked with our application as we enhance it further. LinkedIn job notifications, Email Job notifications- will all available for the user to add to the Wish-List.

Additional functionality can include:

1. Having Portal link at which the individual jobs can be applied.
2. Resume storage facility per job description
3. Timely reminders based on application deadline dates
4. Suggestions on available opportunity based on previous history

3 KERNEL BEST PRACTICES FOLLOWED

The overall project was built following the best practices in software development.

Below are listed all the kernel best practices and it's implementation with examples in the defined project:

3.1 Zero Internal Boundaries

We ensured that everyone in the team has access to all the tools being used. Most of the back-end was coded using Python (PyCharm) and the team members were comfortable using the same.

The front-end of the application was developed using React.js and node.js that was compatible with the systems used by all team members.

Since the complete project was built using Python and basic knowledge of JavaScript everyone in the team was able to quickly pickup and complete their own pieces.

3.2 Short Release Cycle

The overall release cycle focused on delivering MVP version of the application and hence there were no shorter releases in the window of 4- weeks. Almost all codes were pushed in a span of 2-3 days and a few bug fixes were pushed after a testing cycle.

3.3 Distributed Development Model

The overall work was divided well amongst all team members considering the everyone's schedules. The commits on the git-hub also reflect the same. Once the project idea was brainstormed and aligned upon, two of us heavily worked on front-end and the rest three on back-end and documentation. However, everyone was aware of the statuses and progress across all the divisions.

3.4 Consensus Oriented Model

There were several discussions and brainstorm meetings that happened to align on the overall project idea, the outcomes that we wanted to achieve and overall design we wanted to follow.

As we progresses through developing the system, all the issues, uncertainties and challenges were discussed, opinions were heard and aligned decisions were made.

3.5 No Regression Rule

Currently, the scope does not have any issues that would require regression testing. The future scope however does define various modules / additional functionality that would require regression testing of the complete system.

REFERENCES

- [Gre93] George D. Greenwade. The Comprehensive Tex Archive Network (CTAN). *TUGBoat*, 14(3):342–351, 1993.