WolfJobs

The Ultimate Hub for On-Campus Student Employment

GROUP 73

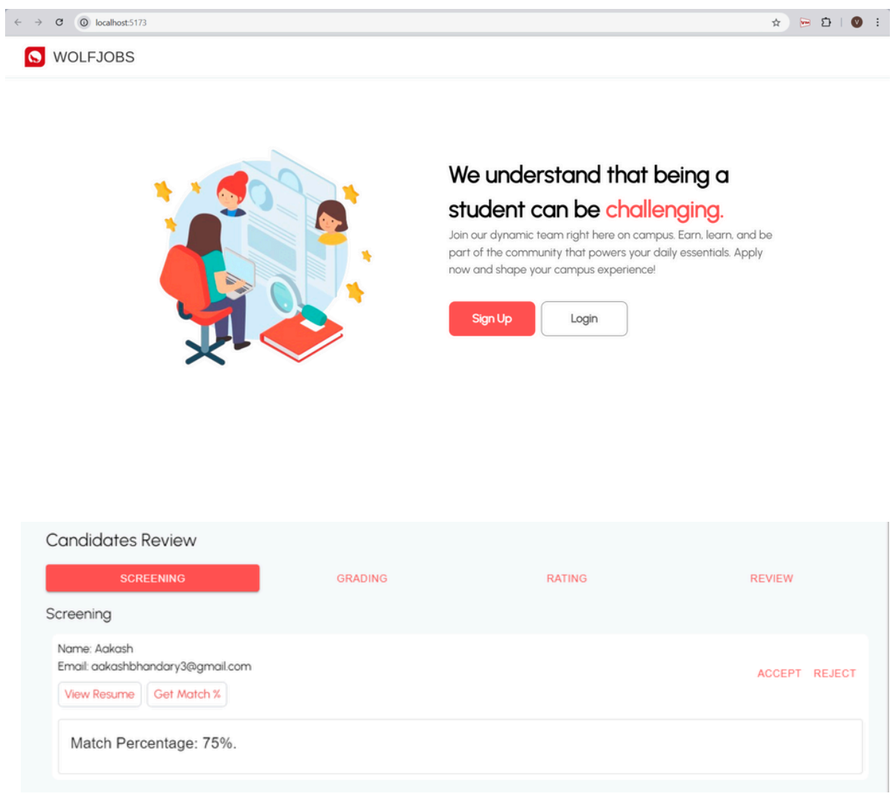
Priyanshu | Vaishvi | Aakash
pmalavi | vshah24 | avbhanda

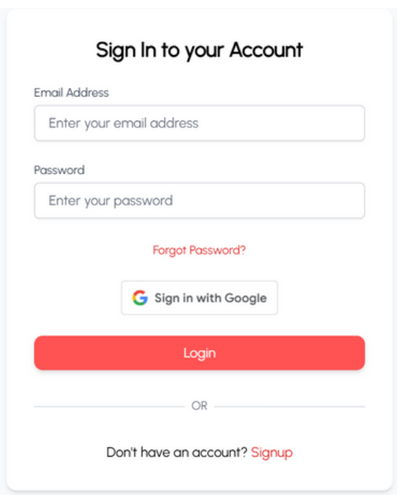
INTRODUCTION

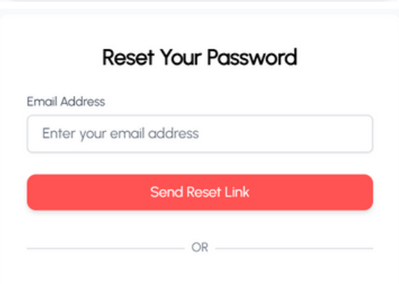
- Welcome to WolfJobs, a one-stop employment platform created to connect NCSU students with flexible on-campus job opportunities while enabling employers to easily access top talent.
- Students can sign up, create personalized profiles, browse job listings, submit applications, and track their status in real time, upload resume, check ATS score ensuring a smooth and transparent job search experience.
- For managers, the platform provides tools to post jobs, edit profiles, screen candidates, and review applications with efficiency.
- The platform's intelligent matching system enhances the process, offering applicants recommendations for jobs that align with their qualifications and allowing managers to quickly find candidates with the right skills.

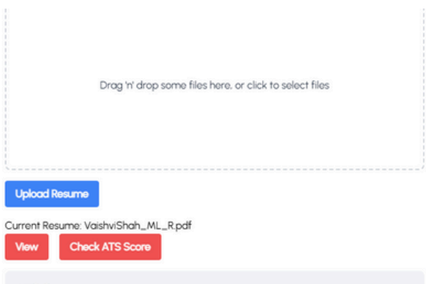
IMPLEMENTED FEATURES

- **Authentication Flows & Security:** Google sign-in simplifies access, while encrypted passwords ensure secure storage and include forgot and reset options for smooth recovery.
- **Email Notification System:** Students receive instant email alerts when their application status updates, allowing them to stay informed about acceptance or rejection decisions.
- **Resume Parsing & ATS Scoring:** The platform's ATS system parses resumes, calculating a precise match score between candidates' skills and job requirements; students can also view their own ATS scores for self-assessment.
- **Show Resume Match % for Managers:** Managers benefit from a clear match percentage based on resume keywords versus job descriptions, helping them quickly identify strong candidates.
- **Profile Setup for Students and Managers:** Profiles include NCSU and student IDs, offering a complete and verified identity for all users.
- **Comprehensive Application Management:** Managers can screen, compare, and select candidates efficiently, with all relevant tools in one centralized platform.









TEST CASES: 65 test cases passed with frontend, backend and API checks

FUTURE ENHANCEMENTS

- **Persistent Sessions:** Improved state management to maintain user sessions and prevent loss upon page refresh.
- **Targeted Notifications:** Customized updates to provide applicants with personalized information regarding their application status.
- **Expanded Application Flexibility:** Allow multiple applications for a single job posting, giving students greater flexibility in their job search.
- **Dynamic Questionnaires:** Implement job-specific questions designed to better evaluate applicant suitability for various roles.
- **Additional Document Uploads:** Enable students to submit cover letters and schedules along with their resumes for a more comprehensive application.
- **Resume Keyword Insights:** Provide feedback on missing keywords in resumes to help applicants enhance their ATS scores and improve their chances of selection.

TECH STACK

M E R N

REST API

GITHUB REPO

DEPLOYED APP

2. Repository Link
https://github.com/SE-Fall24-G73/WolfJobsV2

3.

Rubric	**Self-Assessment Sum: 102**	**Evidence**
Workload is spread over the whole team (one team member is often Xtimes more productive than the others…	3	https://github.com/orgs/SE-Fall24-G73/people
but nevertheless, here is a track record that everyone is contributing a lot)	3	https://github.com/SE-Fall24-G73/WolfJobsV2
Number of commits	3	https://github.com/SE-Fall24-G73/WolfJobsV2
Number of commits: by different people	3	https://github.com/SE-Fall24-G73/WolfJobsV2
Issues reports: there are **many**	3	https://github.com/SE-Fall24-G73/WolfJobsV2/issues
Issues are being closed	3	https://github.com/SE-Fall24-G73/WolfJobsV2/issues
Docs: doco generated, format not ugly	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Docs: what: point descriptions of each class/function (in isolation)	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that makes you want to rush out and use the thing	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Docs: short video, animated,	3	https://github.com/SE-Fall24-

hosted on your repo. That convinces people why they want to work on your code.		G73/WolfJobsV2/blob/master-fall24-g73/README.md
Use of version control tools	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/
Test cases exist	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/
Test cases are routinely executed	3	
Issues are discussed before they are closed	3	https://github.com/SE-Fall24-G73/WolfJobsV2/issues
Chat channel: exists	3	https://github.com/SE-Fall24-G73/WolfJobsV2/issues
Test cases: a large proportion of the issues related to handling failing cases.	2	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/
Evidence that the whole team is using the same tools: everyone can get to all tools and files	3	In GH
Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	3	In GH
Evidence that the whole team is using the same tools (e.g. tutor can ask anyone to share screen, they demonstrate the system running on their computer)	3	In GH
Evidence that the members of the team are working across multiple places in the code base	3	In GH
Short release cycles	2	In GH
The file .gitignore lists what	3	In GH

files should not be saved to the repo. See [examples]i(https://github.com/github/gitignore)		
The file INSTALL.md lists how to install the code	3	No separate install required
The file LICENSE.md lists rules of usage for this repo	3	In GH
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	3	In GH
The file CONTRIBUTING.md lists coding standards and lots of tips on how to extend the system without screwing things up; e.g. see example	3	In GH
The file README.md contains all the following	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Video	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
DOI badge: exists. To get a Digitial Object Indentifier, regiser the project at Zenodo. DOI badges look like this:	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Badges showing your style checkers	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Badges showing your code formatters.	2	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Badges showing your syntax checkers.	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md
Badges showing your code coverage tools	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md

Badges showing any other Other automated analysis tools	3	https://github.com/SE-Fall24-G73/WolfJobsV2/blob/master-fall24-g73/README.md

Online Sustainability

Question	Response	Comment
Question 1.1: Does your website and documentation provide a clear, high-level overview of your software?	Yes	
Question 1.2: Does your website and documentation clearly describe the type of user who should use your software?	Yes	
Question 1.3: Do you publish case studies to show how your software has been used by yourself and others?	Yes	In README
Question 2.1: Is the name of your project/software unique?	Yes	
Question 2.2: Is your project/software name free from trademark violations?	Yes	
Question 3.1: Is your software available as a package that can be deployed without building it?	Yes	No build required
Question 3.2: Is your software available for free?	Yes	
Question 3.3: Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository?	Yes	
Question 3.4: Is your	Yes	On GitHub

software hosted in an established, third-party repository likeGitHub (https://github.com), BitBucket (https://bitbucket.org),LaunchPad (https://launchpad.net) orSourceForge (https://sourceforge.net)?		
Question 4.1: Is your documentation clearly available on your website or within your software?	Yes	
Question 4.2: Does your documentation include a "quick start" guide, that provides a short overview of how to use your software with some basic examples of use?	Yes	
Question 4.3: If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software?	N/A	
Question 4.4: Do you provide a comprehensive guide to all your software's commands, functions and options?	Yes	
Question 4.5: Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for problems and error messages?	No	
Question 4.6: If your software can be used as a	N/A	

library, package or service by other software, do you provide comprehensive API documentation?		
Question 4.7: Do you store your documentation under revision control with your source code?	Yes	
Question 4.8: Do you publish your release history e.g. release data, version numbers, key features of each release etc. on your web site or in your documentation?	Yes	
Question 5.1: Does your software describe how a user can get help with using your software?	Yes	In README
Question 5.2: Does your website and documentation describe what support, if any, you provide to users and developers?	Yes	
Question 5.3: Does your project have an e-mail address or forum that is solely for supporting users?	No	
Question 5.4: Are e-mails to your support e-mail address received by more than one person?	No	
Question 5.5: Does your project have a ticketing system to manage bug reports and feature requests?	Yes	We have an issues page

Question 5.6: Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?	Yes	The issues page is publicly available
Question 6.1: Is your software's architecture and design modular?	Yes	
Question 6.2: Does your software use an accepted coding standard or convention?	Yes	
Question 7.1: Does your software allow data to be imported and exported using open data formats?	No	
Question 7.2: Does your software allow communications using open communications protocols?	Yes	
Question 8.1: Is your software cross-platform compatible?	Yes	
Question 9.1: Does your software adhere to appropriate accessibility conventions or standards?	Yes	
Question 9.2: Does your documentation adhere to appropriate accessibility conventions or standards?	Yes	
Question 10.1: Is your source code stored in a repository under revision control?	Yes	Used GitHub

Question 10.2: Is each source code release a snapshot of the repository?	Yes	
Question 10.3: Are releases tagged in the repository?	Yes	
Question 10.4: Is there a branch of the repository that is always stable? (i.e. tests always pass, code always builds successfully)	Yes	The main branch
Question 10.5: Do you back-up your repository?	No	
Question 11.1: Do you provide publicly-available instructions for building your software from the source code?	N/A	
Question 11.2: Can you build, or package, your software using an automated tool?	No	
Question 11.3: Do you provide publicly-available instructions for deploying your software?	Yes	
Question 11.4: Does your documentation list all third-party dependencies?	Yes	In packages.json in both frontend and backend
Question 11.5: Does your documentation list the version number for all third-party dependencies?	Yes	
Question 11.6: Does your	Yes	

software list the web address, and licences for all third-party dependencies and say whether the dependencies are mandatory or optional?		
Question 11.7: Can you download dependencies using a dependency management tool or package manager?	Yes	
Question 11.8: Do you have tests that can be run after your software has been built or deployed to show whether the build or deployment has been successful?	Yes	
Question 12.1: Do you have an automated test suite for your software?	Yes	
Question 12.2: Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code?	No	
Question 12.3: Do you use continuous integration, automatically running tests whenever changes are made to your source code?	Yes	
Question 12.4: Are your test results publicly visible?	Yes	
Question 12.5: Are all manually-run tests documented?	No	

Question 13.1: Does your project have resources (e.g. blog, Twitter, RSS feed, Facebook page, wiki, mailing list) that are regularly updated with information about your software?	No	
Question 13.2: Does your website state how many projects and users are associated with your project?	Yes	
Question 13.3: Do you provide success stories on your website?	No	
Question 13.4: Do you list your important partners and collaborators on your website?	No	
Question 13.5: Do you list your project's publications on your website or link to a resource where these are available?	No	
Question 13.6: Do you list third-party publications that refer to your software on your website or link to a resource where these are available?	No	
Question 13.7: Can users subscribe to notifications to changes to your source code repository?	No	
Question 13.8: If your software is developed as an open source project (and, not just a project developing open source	No	

software), do you have a governance model?		
Question 14.1: Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project?	Yes	
Question 14.2: Do you have a contributions policy?	Yes	
Question 14.3: Is your contributions' policy publicly available?	Yes	
Question 14.4: Do contributors keep the copyright/IP of their contributions?	Yes	
Question 15.1: Does your website and documentation clearly state the copyright owners of your software and documentation?	Yes	
Question 15.2: Does each of your source code files include a copyright statement?	No	
Question 15.3: Does your website and documentation clearly state the license of your software?	Yes	
Question 15.4: Is your software released under an open source license?	Yes	

Question 15.5: Is your software released under an OSI-approved open-source license?	Yes	
Question 15.6: Does each of your source code files include a license header?	No	
Question 15.7: Do you have a recommended citation for your software?	Yes	
Question 16.1: Does your website or documentation include a project roadmap (a list of project and development milestones for the next 3, 6 and 12 months)?	Yes	
Question 16.2: Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed?	No	
Question 16.3: Do you make timely announcements of the deprecation of components, APIs, etc.?	No	