

**Title:** Job Recruitment Information System

**Author:** Nimish Mathur

**Application Description:** Hello and welcome to the Job Recruitment Information System Web Application! The Job Information System was created and designed to help job seekers and job recruiters keep better track of job applications. The job hunting process can be very tedious and there are very few online tools to help people keep track of places they have applied/ received applications from. As part, this application provides both recruiters and job seekers a more effective way to keep track of what companies they have applied to/ received applications from. System administrators have the ability to edit any part of the job application information system process, allowing for additional creation and/ or clean up of data. The Job Recruitment Information System was also created with the hope of eliminating outdated methods like spreadsheets and hand written notes!

**Authentication and Authorization Scheme:** I have created the Job Recruitment Information System to have 5 key users with unique authorization schemes of what the user is able to access and update:

1) user: tester

pass: {iSchoolUI}

Reason: This is the super user of the Job Recruitment Information System. It has access to all CRUD functionalities embedded within the web application.

2) user: sysadmin

pass: {iSchoolUI}

Reason: The sysadmin user was created for an “employee” of the job recruitment information system website to maintain the application. They have the same root user-like functionality as the superuser, tester.

\*\* The following users have limited access to the information system based on their user permissions:

3) user: schedulingExec

pass: {iSchoolUI}

(able to create company and positions)

Reason: The schedulingExec user was created to help create new job positions for companies. This allows other administrators to more directly add job positions for companies and their associated positions.

4) user: seeker

pass: {iSchoolUI}

Reason: The seeker user was created to view the application system. It doesn't have the ability to add, change or delete anything. It allows job seekers the ability to view the system and job applications they may have applied to (for tracking purposes).

- 5) user: coworker  
pass: {iSchoolUI}  
Reason: Created using a cloning technique from the EZU web application, the coworker only has access to the about page. It was created for someone who may be working on a different project within the same system.

**Tools/ Requirements:** This application conforms with the Anaconda environment used throughout IS 439 (*e4\_trainor\_django\_course*). Nevertheless, I have included the specific environment specs used:

Package	Version
asgiref	3.5.2
bzip2	1.0.8
ca-certificates	2023.01.10
certifi	2022.12.7
Django	4.1
libffi	3.4.2
ncurses	6.3
openssl	1.1.1s
pip	22.3.1
python	3.10.9
readline	8.2
setuptools	65.6.3
sqlite	3.40.1
sqlparse	0.4.3
tk	8.6.12
tzdata	2022g
wheel	0.37.1
xz	5.2.10
zlib	1.2.13

**Testing Instructions:** Running the application is rather straightforward. After cloning and setting the environment up correctly to run python 3.10 and django 4.1, you can run the application using `manage.py runserver`. This will take you to an about page with a login button. To access the root user with all permissions, you can login using:

user: tester  
pass: {iSchoolUI}

\*You can also login with any of the users defined in **Authentication and Authorization Scheme** to test specific authentication and authorization schemes.

Once logged in, you will be able to view the application and its associated populated data. Like the EZU application, the pagination feature is shown using groups of individuals. I used a similar cloning approach to get a large list of job recruiters and job seekers. This in turn has a button at the bottom of the page that allows you to go back and forth between the different pages of job seekers and recruiters.

From the root user perspective, you will also have access to perform CRUD tasks with buttons guiding you through the process (ex: Add a position with the correct corresponding fields).

**Other Relevant Information / Optional Exercise:** I have added test cases to `jobinfo/tests.py`. I have created a full set of test cases that check the validity of the Model classes, Forms, URL patterns, and List Views. To run the test cases, open a manage.py terminal and run `test jobinfo`.

Lastly, I have created a variation of generic class based views and class based views to show the different patterns and the powerful condensation of code patterns under gcbv. I have also tried to highlight the key parts of the django framework using models, forms, views, url configurations, templates, and authentication/ authorization schemes.

I would also like to mention that I do plan to work on this application more in the future. I think there are very few good ways to keep track of job applications as a job seeker actively looking for employment. I want to improve the user interface by adding styling and maybe more functionality like storing a resume and relevant skills for easy access.

Thank you again for taking the time to read this and I hope you enjoy the application!