SmartHire

Team Technovators

1) Dhanasree Rajamani
2) Chaitra Bengaluru Vishweshwaraiah
3) Krishna Pranathi Mokshagundam
4) Sravani Thota

SmartHire Application Instance on AWS

http://13.56.8.55/

SmartHire Application Demo

1) Application Demo : https://youtu.be/KFXla2sy6zY

2) Getting resumes from S3, parsing and displaying on the application : https://drive.google.com/file/d/1bsTYfK_ghJF7bJusPGghHBsCc9a-v0l3/view?usp=share_link

Github URL - Code

https://github.com/Dhanasree-Rajamani/SmartHire_CMPE272

Problems with Recruitment process

Recruitment is a tedious process that enterprise organizations have to go through to acquire the right talent. The process of recruitment takes a lot of effort, time and cost for the organization with the below mentioned concerns

1. Shortlisting the right profiles from a wide pool of applicants without right qualifications
2. Scheduling interviews according to the interviewer and the candidates convenience
3. To figure out how appropriate is the candidate for the role that he/she has applied for

Why SmartHire?

SmartHire is the solution for enterprise organizations to make the recruitment process efficient and effective by automating the functionalities involved in recruitment. It is a one-stop solution where shortlisting the candidates and scheduling the interviews with the candidates can be done easily with a single tool.

It eliminates the need for a third party organization, therefore reducing cost. It is a user-friendly tool for the recruiters to view, shortlist, interview, provide feedback and select/reject the candidate, while ensuring that the data is secure(Access to confidential data is role based).

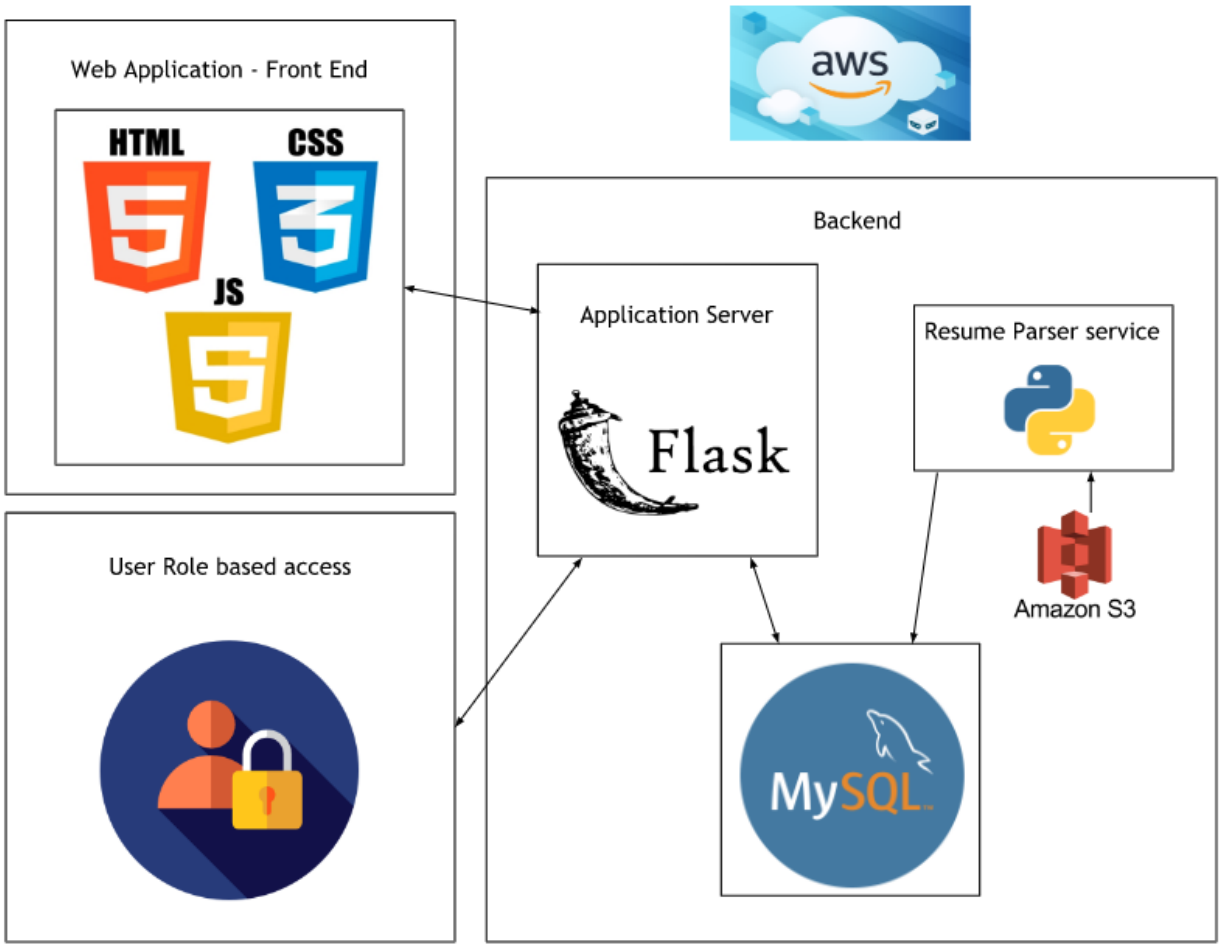
We provide the organization with a feature of scoring the profiles of the candidates based on their skillset and their work experience on that particular skill which makes it easier to shortlist the right candidates from the large pool of candidates.

Functionalities

- ➢ Candidate resumes are downloaded from the cloud.
- ➢ The parser is used to parse the resume downloaded from the cloud to store the required fields in the recruitment database.
- ➢ The list of roles that are currently open for recruitment is displayed, from which relevant candidates for the role are automatically shortlisted.

- ➢ The recruiter can add, view and close a position on the application.
- ➢ The recruiter can view all the candidates as well as candidates relevant to a role. The recruiter can schedule interviews to candidates(application automatically assigns candidates with most relevant interviewer).
- ➢ Candidate resumes can be viewed from within the application.
- ➢ The application also provides a portal to provide and view interview feedback of candidates.

System Architecture



Technologies/Frameworks Used

- **Flask** : It is python microframework used to develop the web application. We have used packages like flask_login, flask_wtf for login management and handling Form requests.

- **MySQL** : MySQL relational database is used to store the applicant data parsed from the resumes into the candidate table. Some of the tables used in MySQL are tbl_employees, tbl_candidate_resume, tbl_open_roles, tbl_shortlisted_candidates etc.

- **Python** : Parsing the resumes and obtaining the candidate details is done using a python library pyreparser. Other functionalities such as shortlisting candidates, assigning a relevancy score, scheduling interviews for candidates with relevant interviewers are also done with Python.

- **Amazon S3** : S3 holds all the resumes of candidates that have applied for the open roles.

- **AWS** : The application has been hosted on AWS EC2.

- **HTML,CSS & Javascript** : Used to develop the web pages and adding several styles using bootstrap CSS.

- **NLTK** : Natural Language Toolkit is a python package used in the pyreparser package for parsing resumes.

Stages

1) Add/View a job role
2) Get all candidates
3) Shortlist relevant candidates for an open position.
4) Get interviewers(from employees) for an open position.
5) Assign relevant interviewers to shortlisted candidates.
6) Feedback gathering stage.
7) Select/ Reject candidate
8) Close a job role

Use Cases

Role based access

We have implemented role based access in SmartHire application. There are two specific roles to gain access namely

- Recruiter
- Employee

The recruiter has more functionalities compared to that of an employee.

The various features of the application are described here, along what role based access has been implemented to those functionalities

Recruiter

Add roles : The Recruiters can post the name, requirements(skill sets and years of experience) and the job description of an open role within the organization

Open roles : View the positions which are available, which will help the recruiter to have a consolidated view of all open roles.

Shortlist candidates : From the list of open job roles, clicking on the application button of a job role shortlists and displays all the candidates that are relevant to the selected job role. This is based on matching the candidate skillset and years of experience with the skills and experience required for the open role. A score is assigned to the candidate based on how relevant they are to a specific role, and the candidates are displayed on the portal in descending order of this score. The shortlisted candidates are displayed, along with their skill set, relevancy score, email etc.

Assign Interview : A recruiter can schedule an interview for a shortlisted candidate. This also can be done on the click of a button. The application automatically gets the list of interviewers for a role(based on matching the skills and work experience of the employees with that of the open job role), and assigns the candidate with the most relevant interviewer.

View all candidates : It provides an overall view of all the candidates who have applied for different positions within the organization, their skillset, work experience, education and contact details.

View all employees : This feature helps the recruiter to have a consolidated view of all the employees within the organization, their skillset, work experience along with their current role.

Post a Feedback : The recruiter can post feedback for a candidate after their interview.

View Feedback : The recruiter can also view feedback of all candidates(provided by employees) based on their interview performances.

Close Job role : The recruiter can close a job role once a candidate has been hired for the role and/or if the role is no longer open for recruitment.

Employees

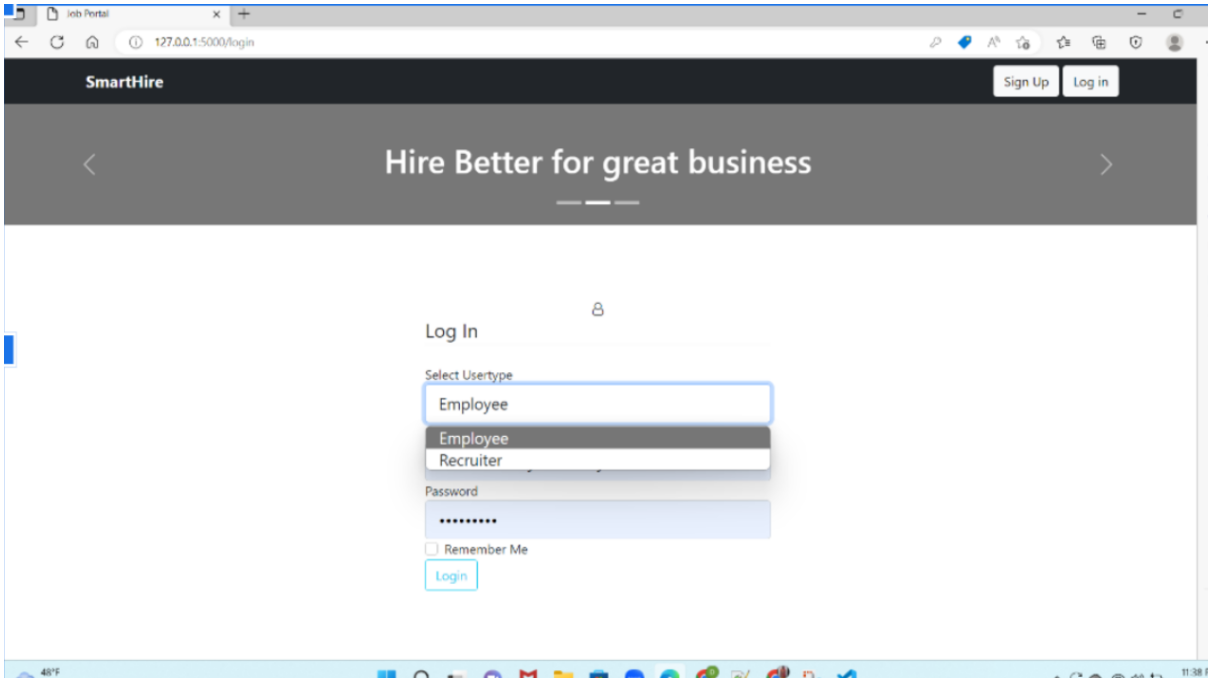
Open roles : The roles which are available along with the information of the shortlisted candidates. The employees can only view the roles and candidates. They cannot add or close job roles, and they cannot shortlist candidates or assign interviews to them.

Post a feedback : An employee can post the feedback of the candidates they interviewed. They cannot view the interview feedback of the candidates.

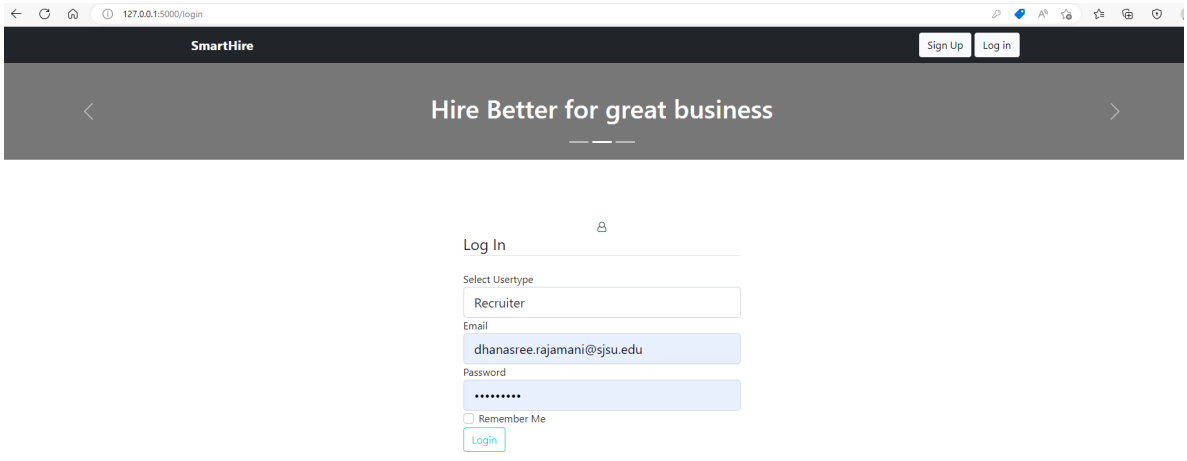
Screenshots of the application

Home page

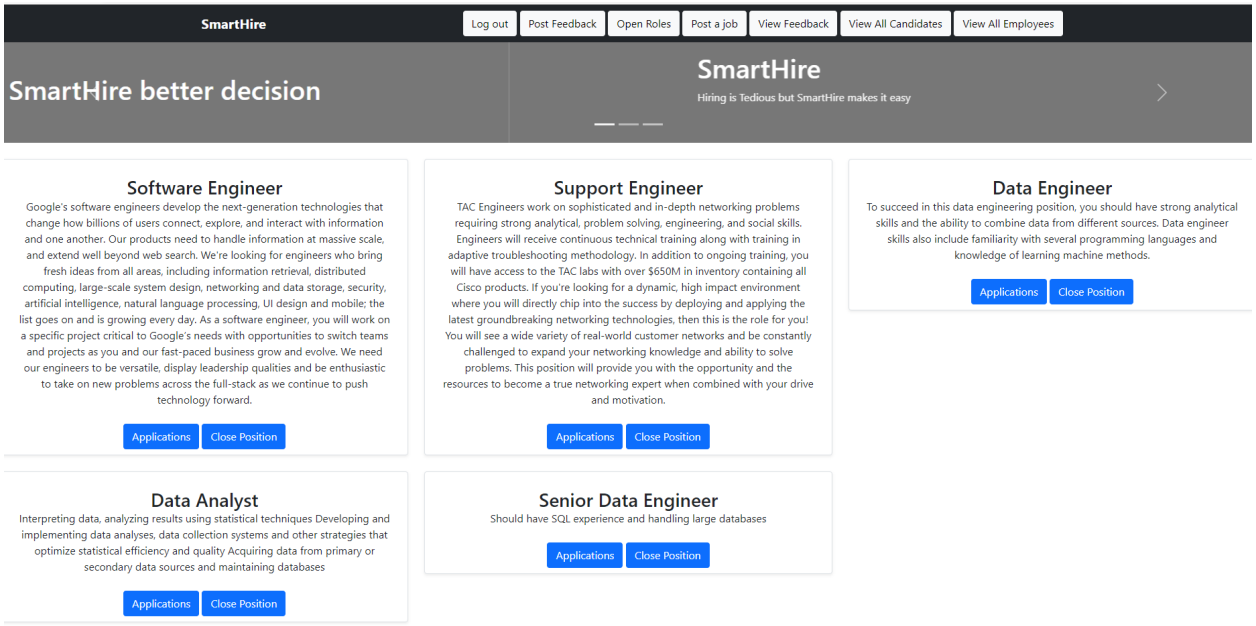
Log in as Employee or Recruiter



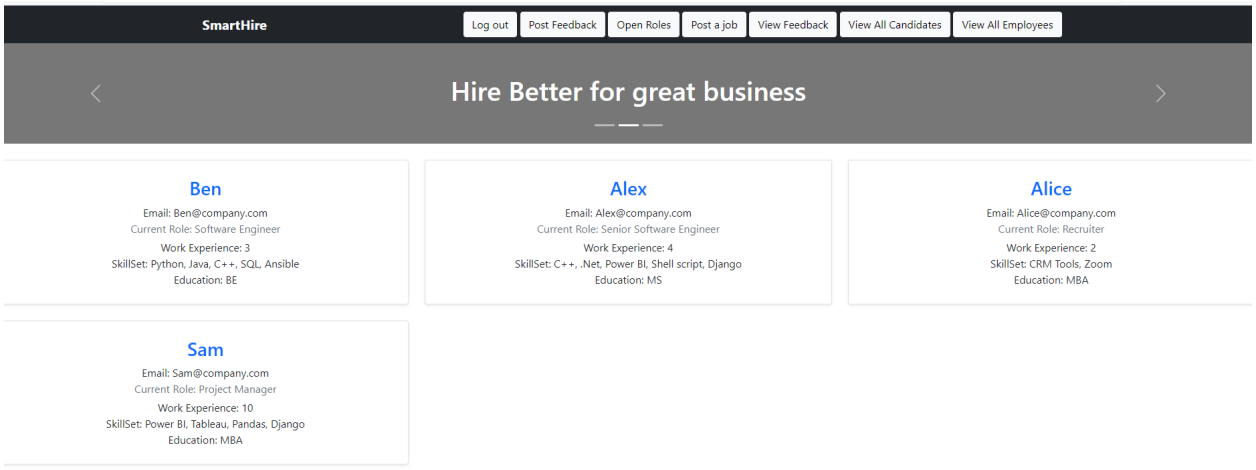
Logging in as Recruiter



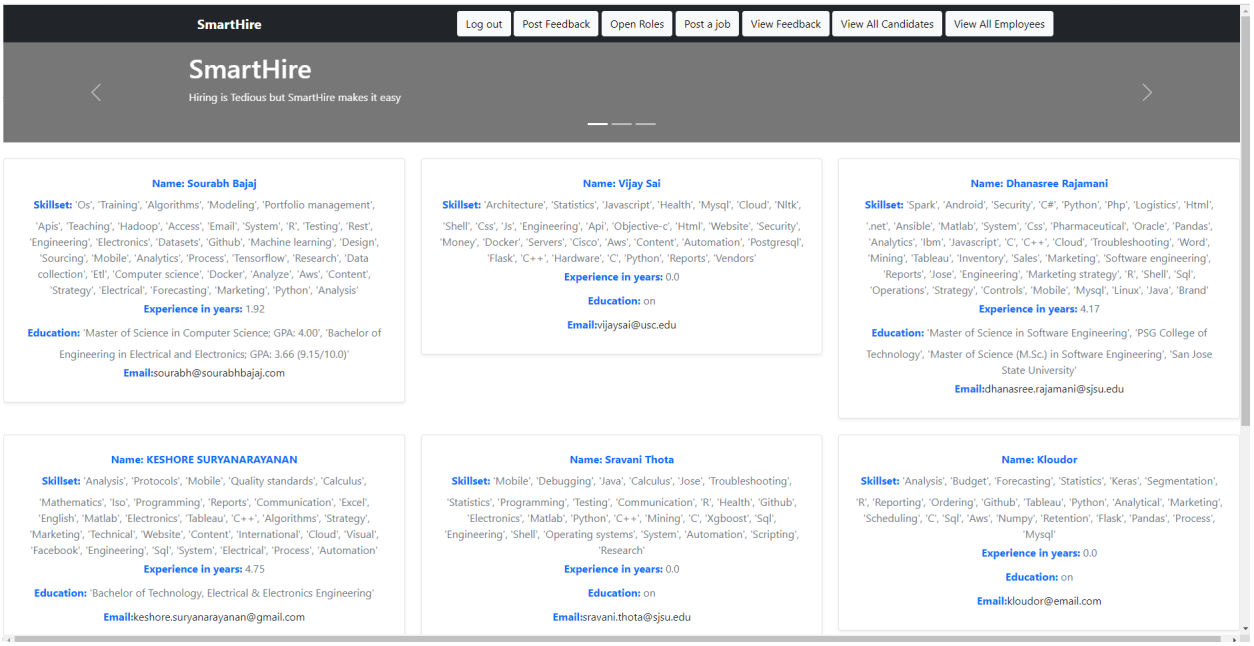
View open job roles



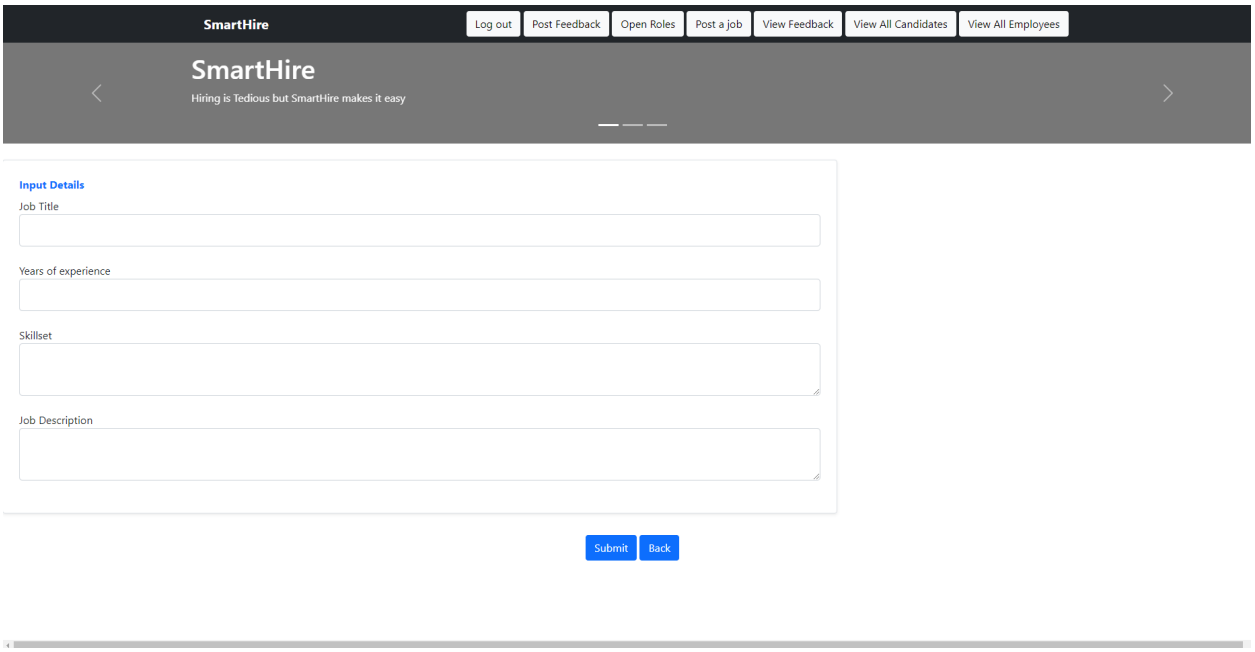
View All employees



View all candidates

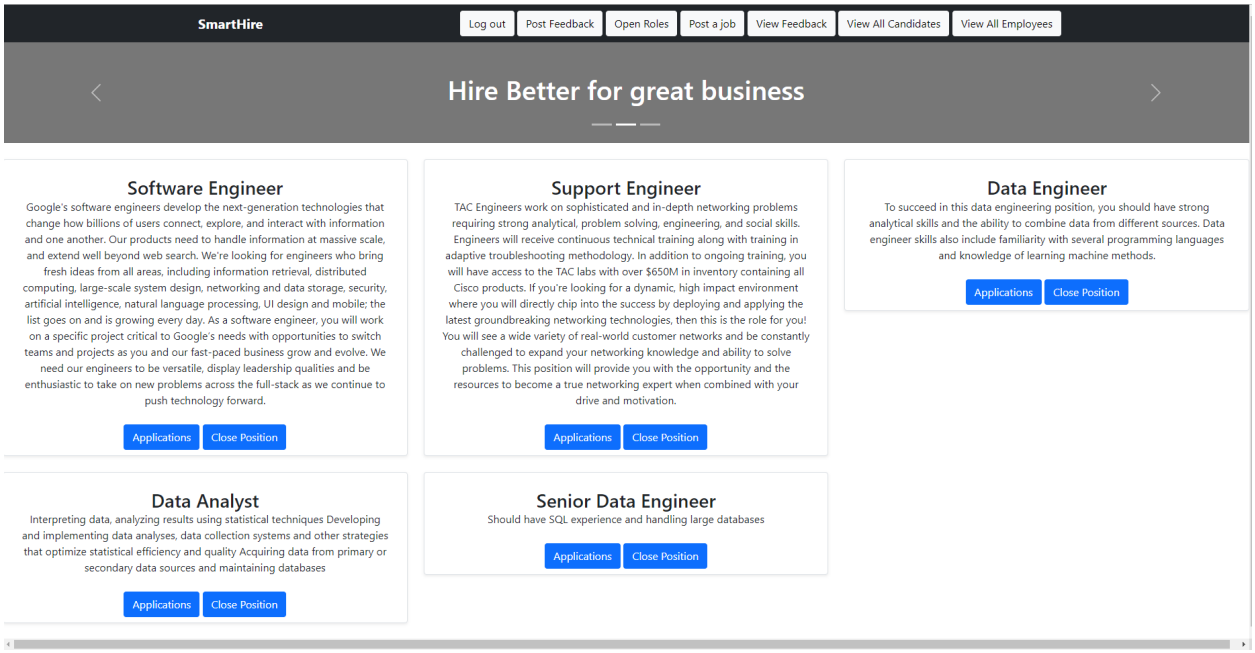


Post a Job opening



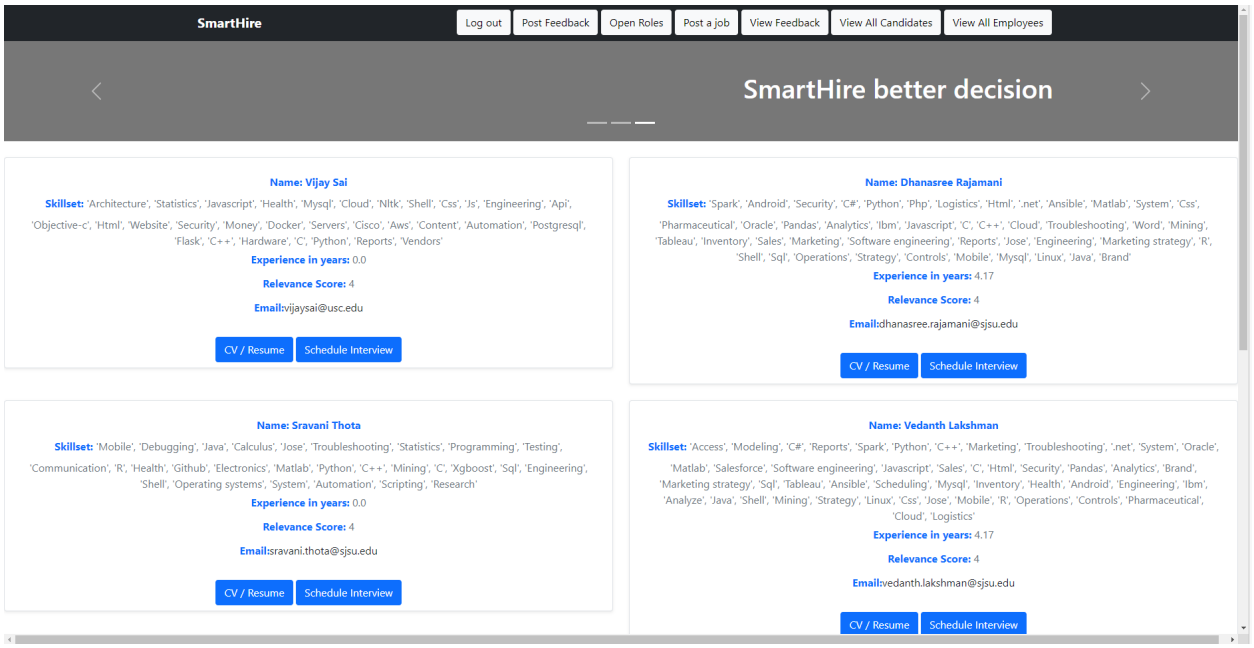
View Open Roles

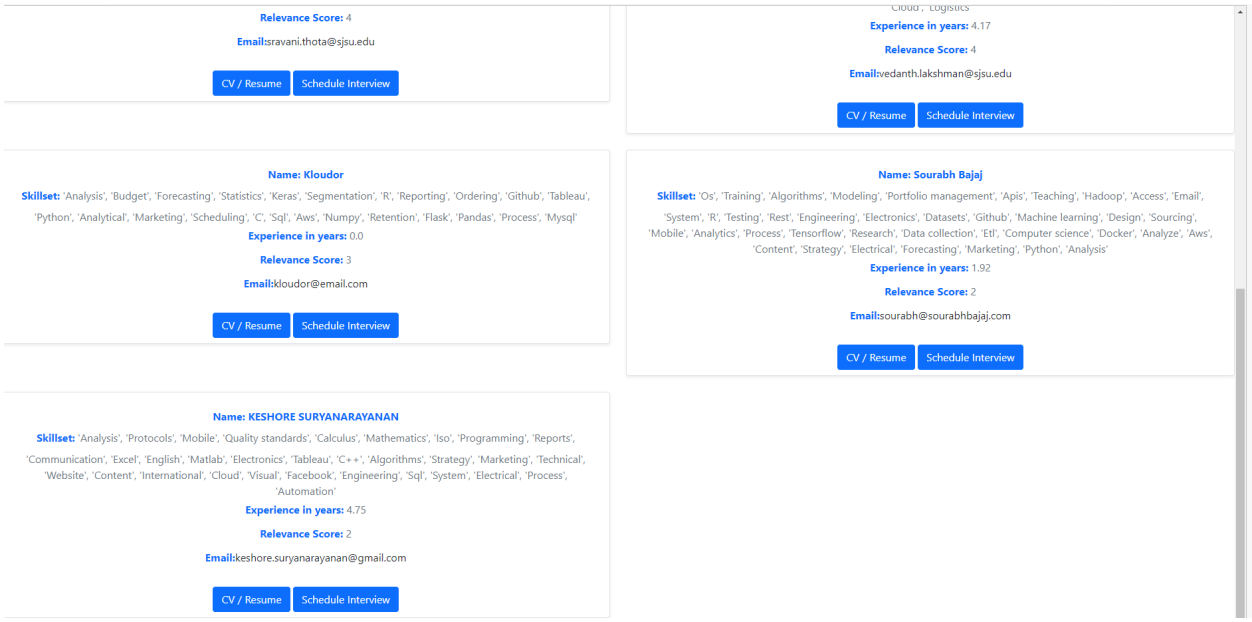
Clicking on applications here, automatically shortlists relevant candidates for the role
Click on close position closes the role - job role no longer open for recruitment.



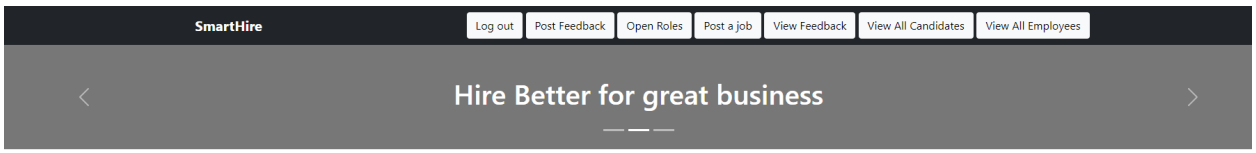
View shortlisted candidates for Software engineer role

We can see candidate details, along with their score calculated by the application

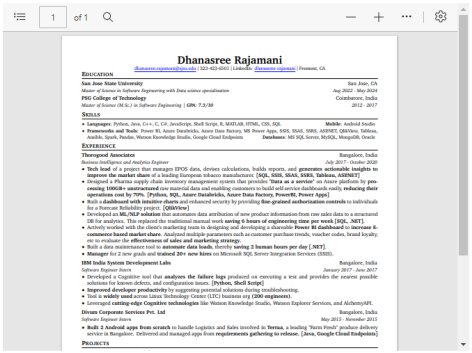




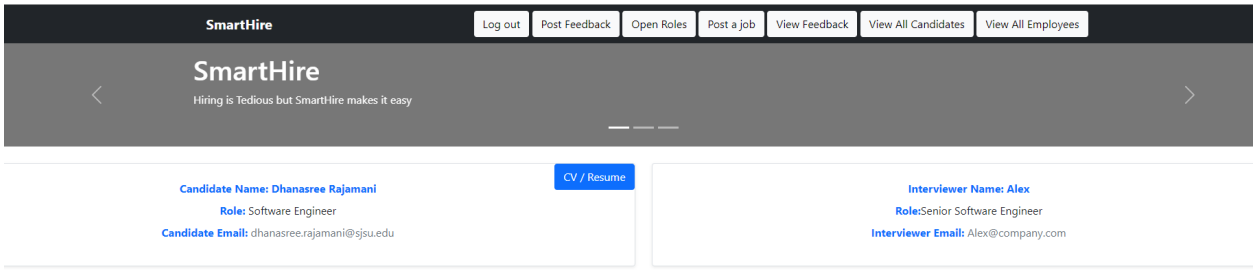
View Shortlisted candidate Resume



CV/Resume

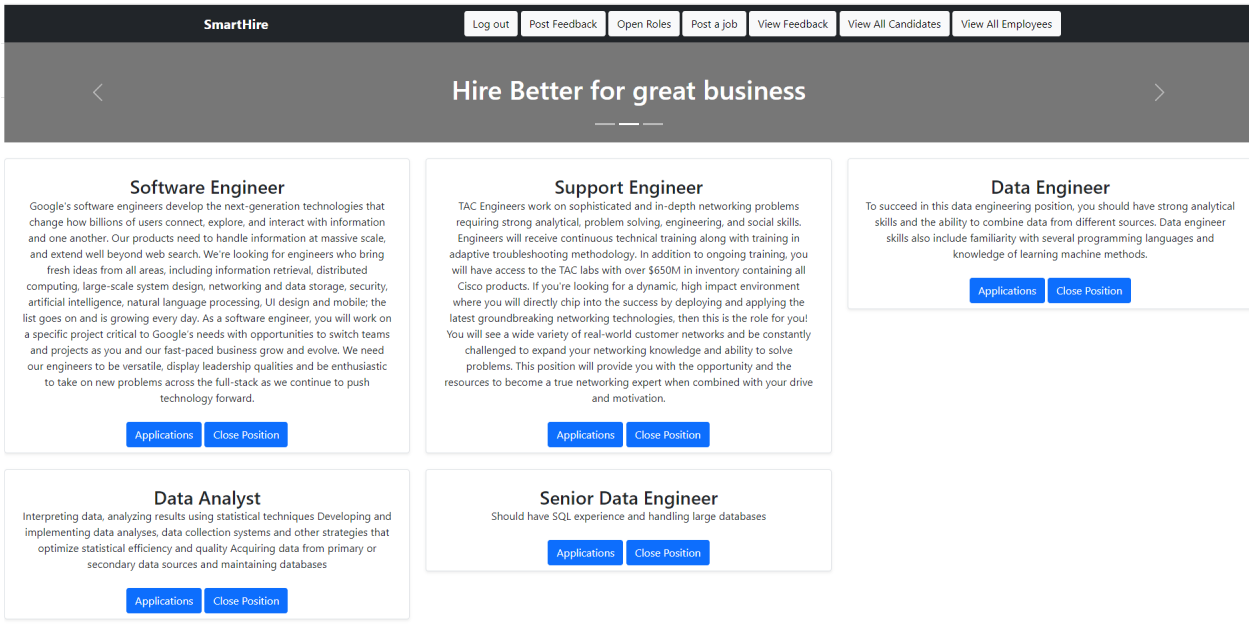


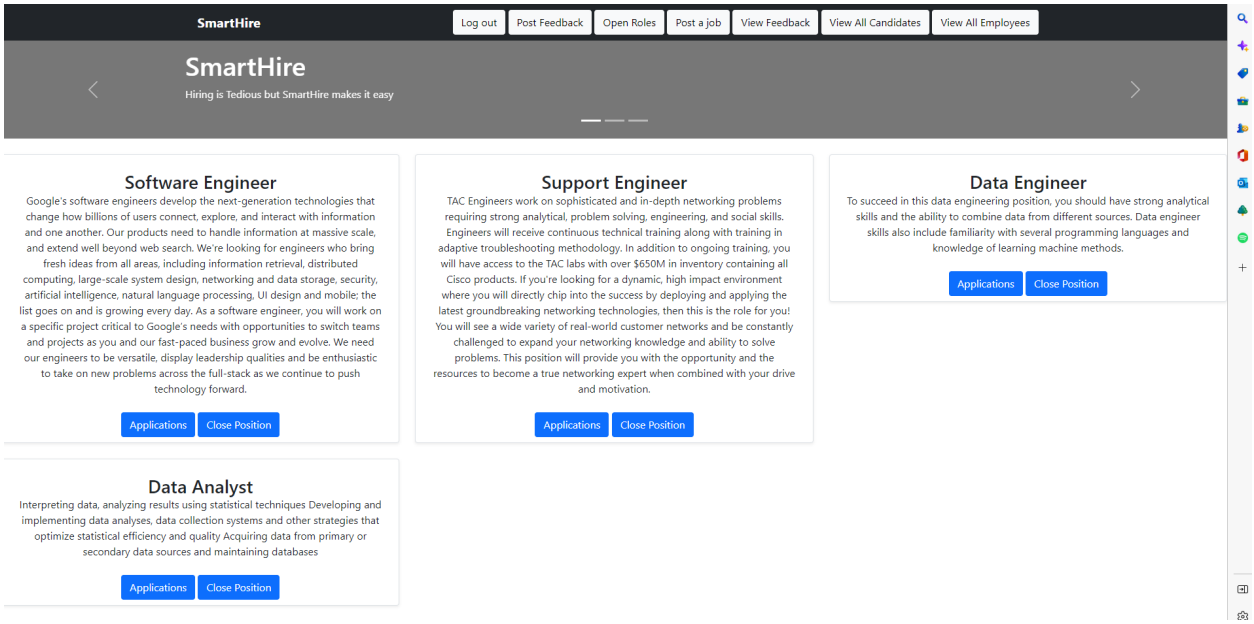
Schedule Interview for a candidate



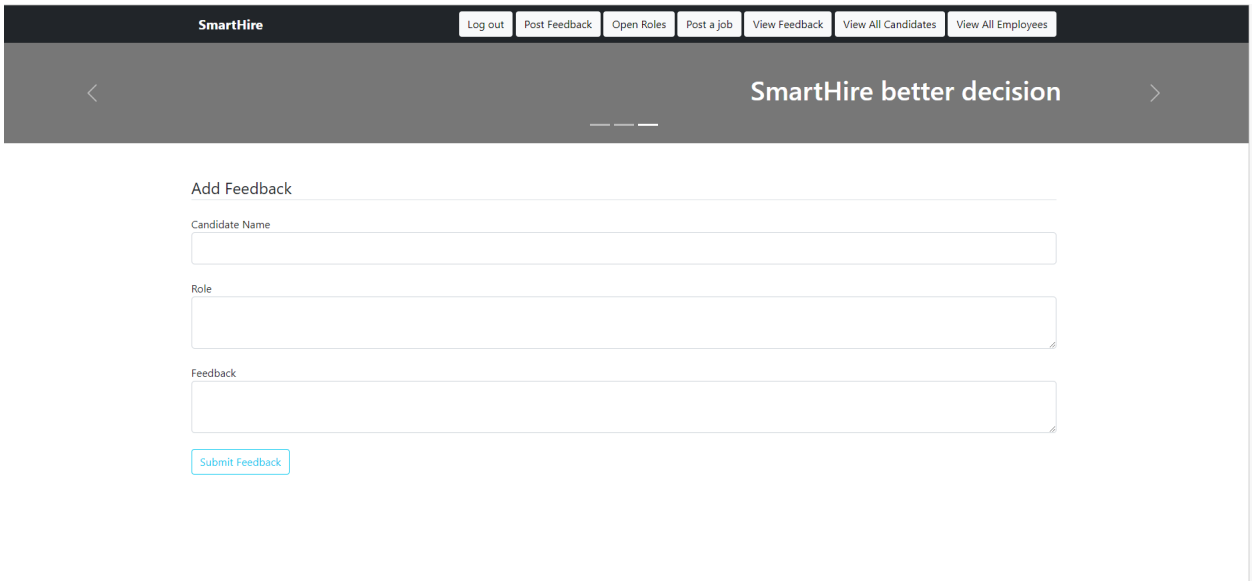
Close a position

Clicking on close position for senior data engineer, it can no longer be viewed

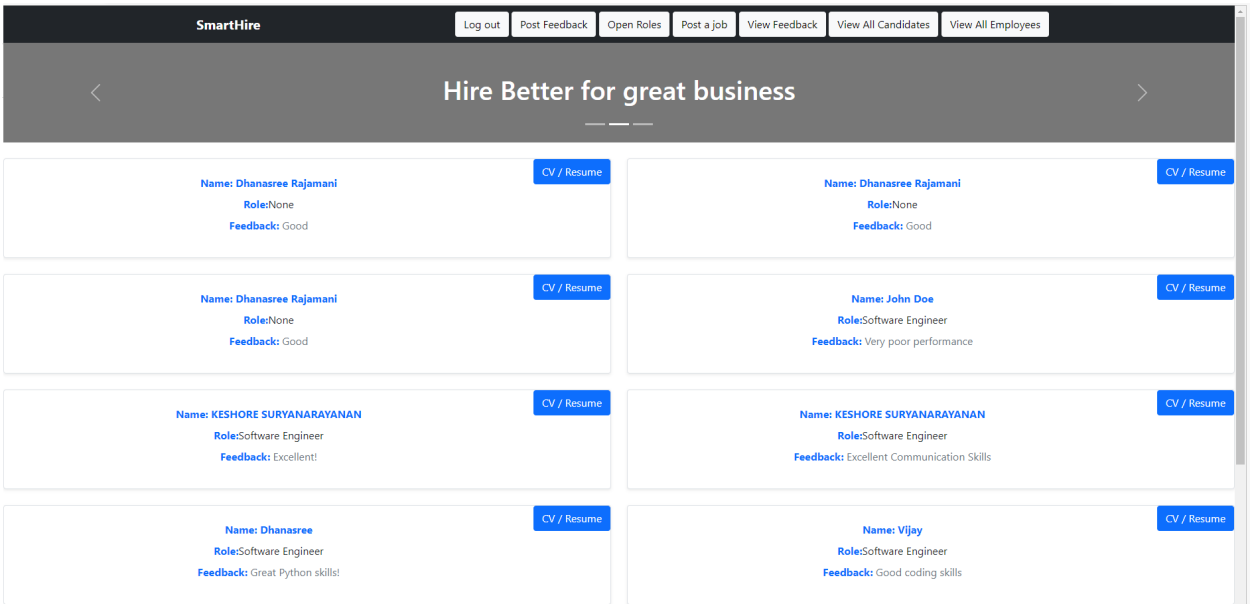




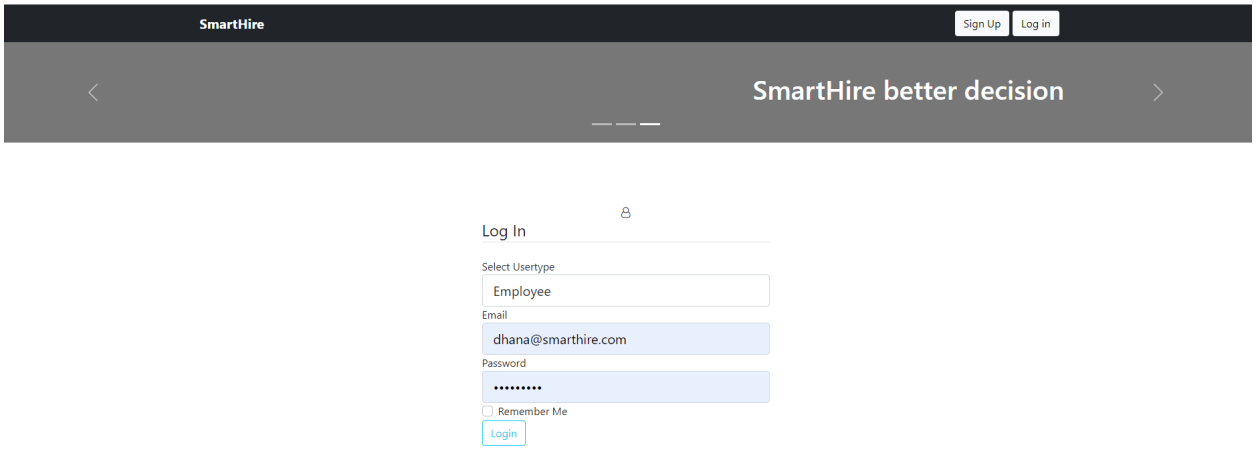
Provide feedback for candidates



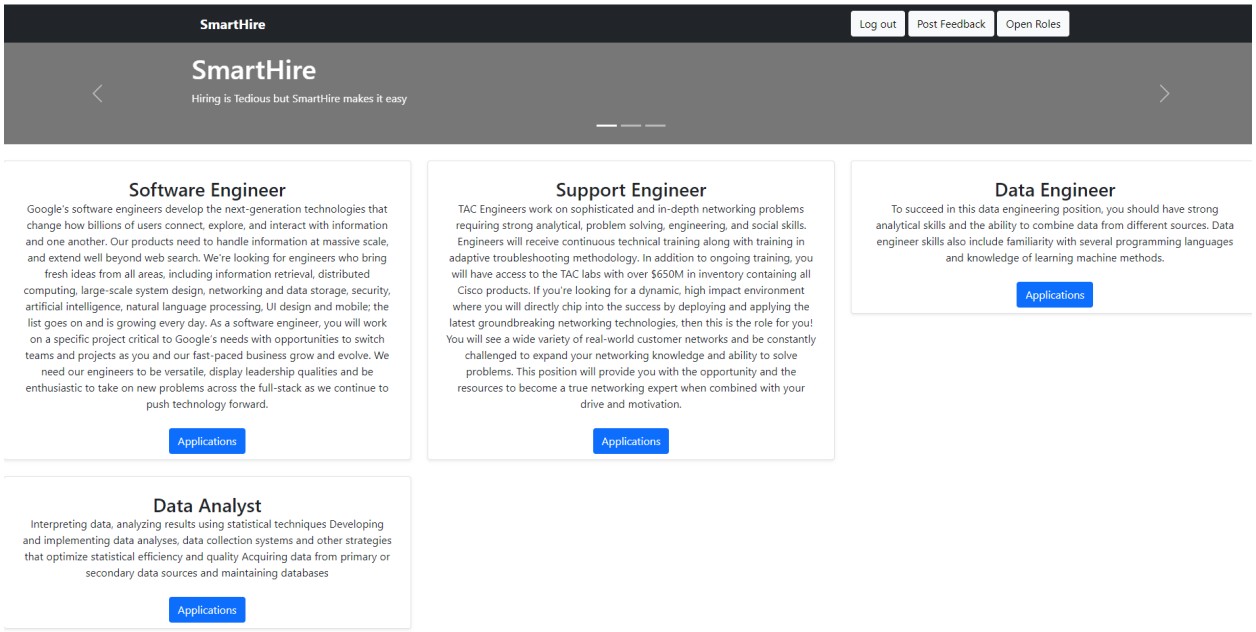
View all candidates Feedback



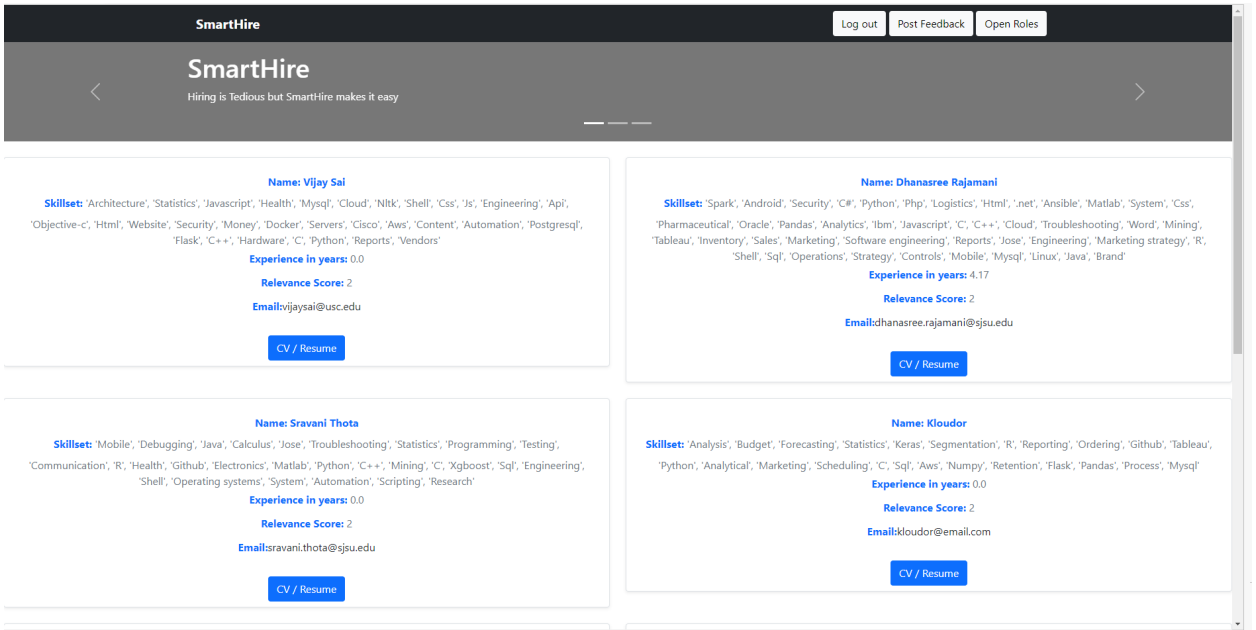
Logging in as Employee



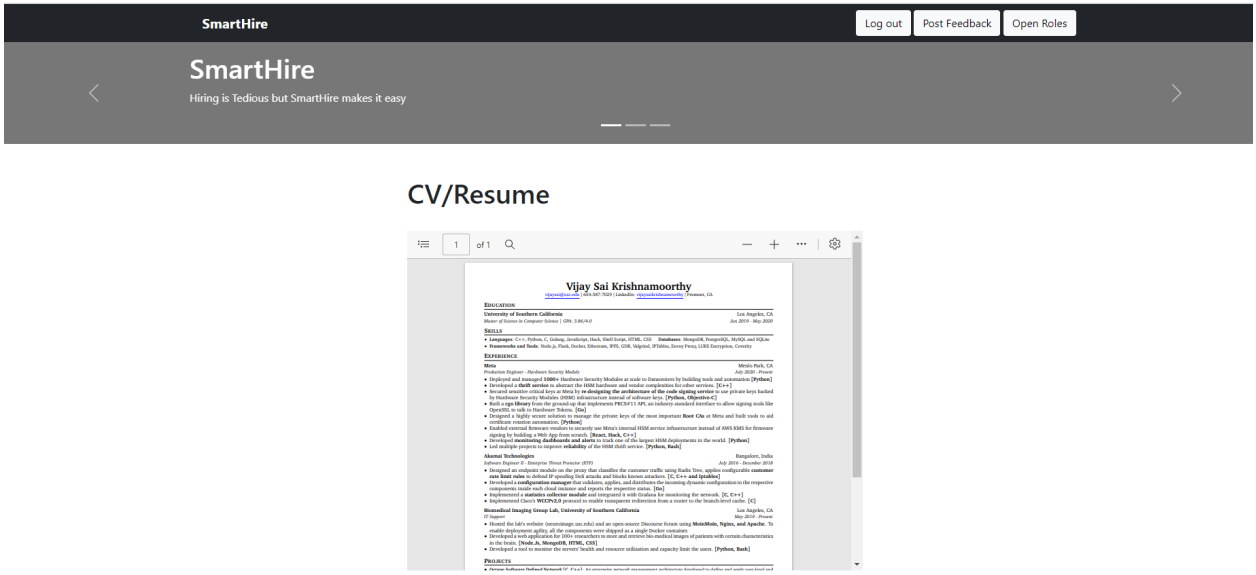
View open roles



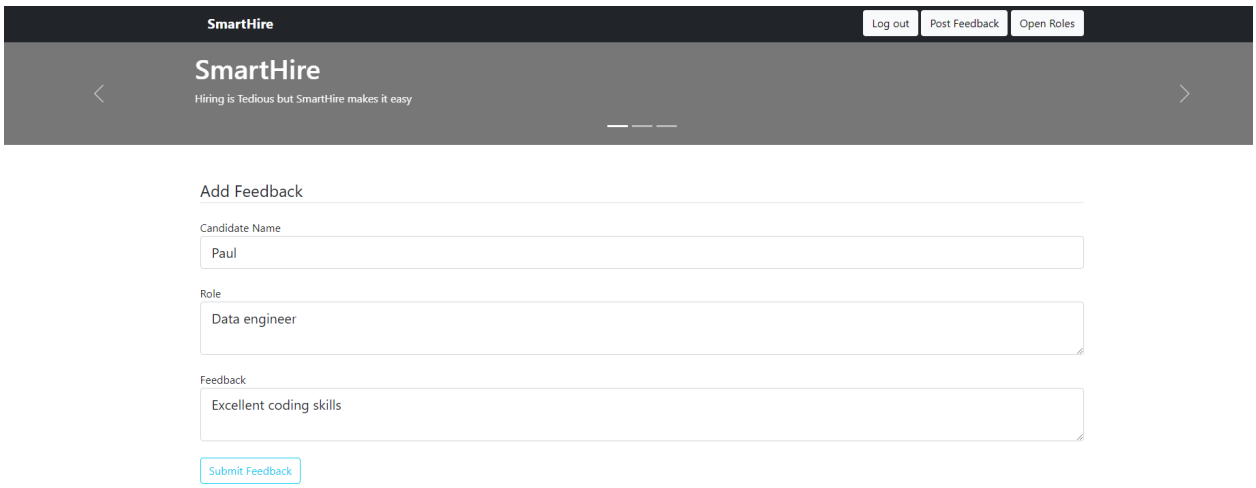
View shortlisted candidates for an open role



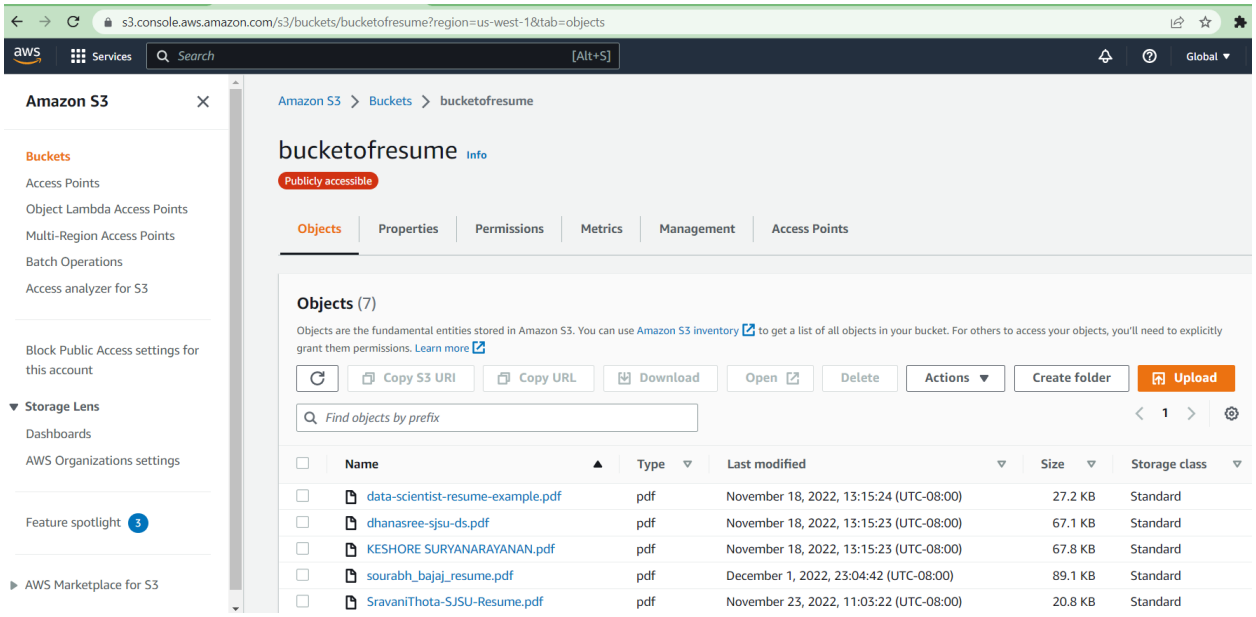
View shortlisted candidates Resume



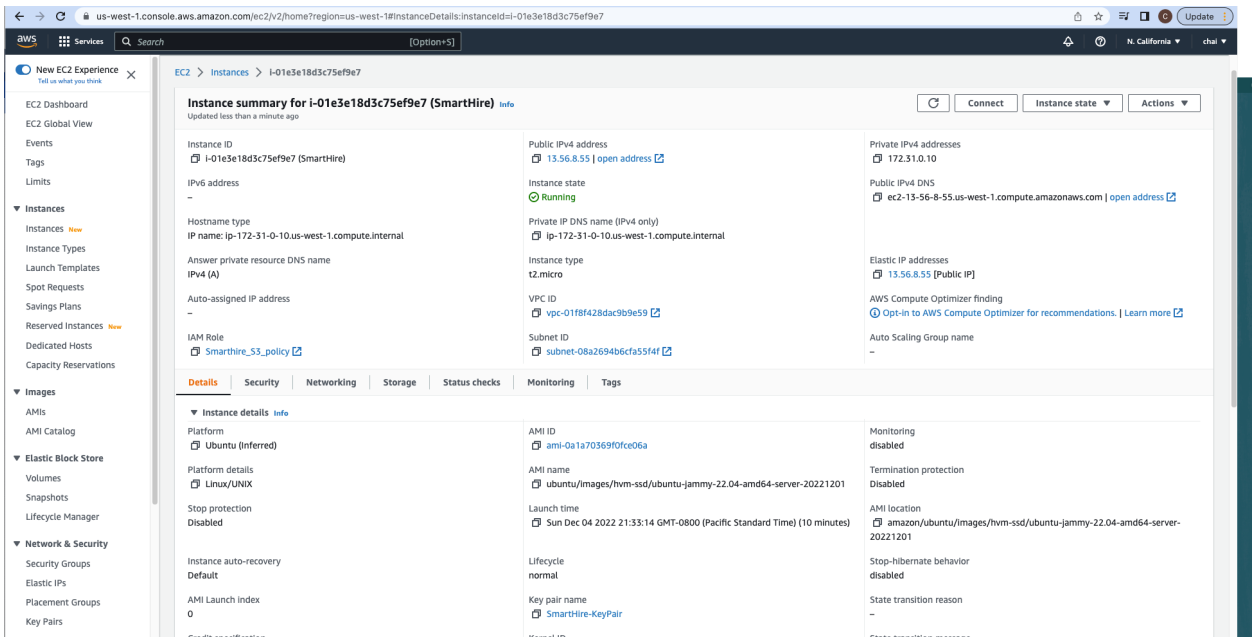
Post feedback



Resumes on S3

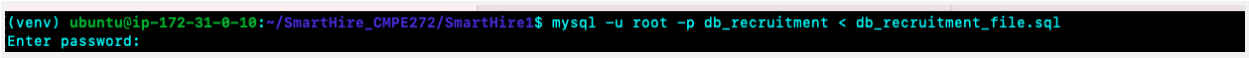


Application Instance on AWS EC2



Deploying a SmartHire application on AWS EC2

1. SSH into Ubuntu EC2
   ssh -i <your key name>.pem ubuntu@<Public DNS of your EC2>.
2. Create/Clone Flask application inside EC2
   Copied smarthire flask application to EC2 instance using scp command
   Installed MYSQL db along with related dependencies and Verified by running python app.py



```
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.31-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use db_recruitment;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from tbl_resume_link;
+----+----------------------------------+-------------+------------------------+
| ID | Resume_link                      | Loaded_flag | Candidate_name         |
+----+----------------------------------+-------------+------------------------+
| 28 | sourabh_bajaj_resume.pdf         |           1 | Sourabh Bajaj          |
| 29 | vijaysai.pdf                     |           1 | Vijay Sai              |
| 30 | dhanasree-sjsu-ds.pdf            |           1 | Dhanasree Rajamani     |
| 31 | KESHORE SURYANARAYANAN.pdf       |           1 | KESHORE SURYANARAYANAN |
| 32 | SravaniThota-SJSU-Resume.pdf     |           1 | Sravani Thota          |
| 33 | data-scientist-resume-example.pdf |           1 | Kloudor                |
| 34 | vedanth_resume.pdf               |           0 | NULL                   |
+----+----------------------------------+-------------+------------------------+
7 rows in set (0.00 sec)
```

```
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1/app$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.31-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use db_recruitment;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+----------------------------+
| Tables_in_db_recruitment   |
+----------------------------+
| tbl_candidate_resume       |
| tbl_employees              |
| tbl_feedback               |
| tbl_open_roles             |
| tbl_resume_link            |
| tbl_shortlisted_candidates |
+----------------------------+
6 rows in set (0.00 sec)
```

3. Run Gunicorn WSGI server to serve the Flask Application

```
^C(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1/app$ pip install gunicorn
Collecting gunicorn
  Downloading gunicorn-20.1.0-py3-none-any.whl (79 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 79.5/79.5 KB 1.9 MB/s eta 0:00:00
Requirement already satisfied: setuptools>=3.0 in /home/ubuntu/SmartHire_CMPE272/SmartHire1/venv/lib/python3.10/site-packages (from gunicorn) (59.6.0)
Installing collected packages: gunicorn
Successfully installed gunicorn-20.1.0
```

```
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ gunicorn -b 0.0.0.0:8000 my_app:smarthire_app --log-level=debug
[2022-12-05 03:00:14 +0000] [5455] [DEBUG] Current configuration:
  config: ./gunicorn.conf.py
  wsgi_app: None
  bind: ['0.0.0.0:8000']
  backlog: 2048
  workers: 1
  worker_class: sync
  threads: 1
  worker_connections: 1000
  max_requests: 0
  max_requests_jitter: 0
  timeout: 30
  graceful_timeout: 30
  keepalive: 2
  limit_request_line: 4094
  limit_request_fields: 100
  limit_request_field_size: 8190
  reload: False
  reload_engine: auto
  reload_extra_files: []
  spew: False
  check_config: False
  print_config: False
  preload_app: False
  sendfile: None
  reuse_port: False
  chdir: /home/ubuntu/SmartHire_CMPE272/SmartHire1
  daemon: False
  raw_env: []
  pidfile: None
  worker_tmp_dir: None
  user: 1000
  group: 1000
  umask: 0
  initgroups: False
  tmp_upload_dir: None
  secure_scheme_headers: {'X-FORWARDED-PROTOCOL': 'ssl', 'X-FORWARDED-PROTO': 'https', 'X-FORWARDED-SSL': 'on'}
  forwarded_allow_ips: ['127.0.0.1']
  accesslog: None
  disable_redirect_access_to_syslog: False
  access_log_format: %(h)s %(l)s %(u)s %(t)s "%(r)s" %(s)s %(b)s "%(f)s" "%(a)s"
  errorlog: -
  loglevel: debug
  capture_output: False
  logger_class: gunicorn.glogging.Logger
  logconfig: None
  logconfig_dict: {}
  syslog_addr: udp://localhost:514
  syslog: False
  syslog_prefix: None
  syslog_facility: user
  enable_stdio_inheritance: False
  statsd_host: None
```

```
  syslog_facility: user
  enable_stdio_inheritance: False
  statsd_host: None
  dogstatsd_tags:
  statsd_prefix:
  proc_name: None
  default_proc_name: my_app:smarthire_app
  pythonpath: None
  paste: None
  on_starting: <function OnStarting.on_starting at 0x7f9ac4516ef0>
  on_reload: <function OnReload.on_reload at 0x7f9ac4517010>
  when_ready: <function WhenReady.when_ready at 0x7f9ac4517130>
  pre_fork: <function Prefork.pre_fork at 0x7f9ac4517250>
  post_fork: <function Postfork.post_fork at 0x7f9ac4517370>
  post_worker_init: <function PostWorkerInit.post_worker_init at 0x7f9ac4517490>
  worker_int: <function WorkerInt.worker_int at 0x7f9ac45175b0>
  worker_abort: <function WorkerAbort.worker_abort at 0x7f9ac45176d0>
  pre_exec: <function PreExec.pre_exec at 0x7f9ac45177f0>
  pre_request: <function PreRequest.pre_request at 0x7f9ac4517910>
  post_request: <function PostRequest.post_request at 0x7f9ac45179a0>
  child_exit: <function ChildExit.child_exit at 0x7f9ac4517ac0>
  worker_exit: <function WorkerExit.worker_exit at 0x7f9ac4517be0>
  nworkers_changed: <function NumWorkersChanged.nworkers_changed at 0x7f9ac4517d00>
  on_exit: <function OnExit.on_exit at 0x7f9ac4517e20>
  proxy_protocol: False
  proxy_allow_ips: ['127.0.0.1']
  keyfile: None
  certfile: None
  ssl_version: 2
  cert_reqs: 0
  ca_certs: None
  suppress_ragged_eofs: True
  do_handshake_on_connect: False
  ciphers: None
  raw_paste_global_conf: []
  strip_header_spaces: False
[2022-12-05 03:00:14 +0000] [5455] [INFO] Starting gunicorn 20.1.0
[2022-12-05 03:00:14 +0000] [5455] [DEBUG] Arbiter booted
[2022-12-05 03:00:14 +0000] [5455] [INFO] Listening at: http://0.0.0.0:8000 (5455)
[2022-12-05 03:00:14 +0000] [5455] [INFO] Using worker: sync
[2022-12-05 03:00:14 +0000] [5456] [INFO] Booting worker with pid: 5456
[2022-12-05 03:00:14 +0000] [5455] [DEBUG] 1 workers
[nltk_data] Downloading package stopwords to /home/ubuntu/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package stopwords to /home/ubuntu/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Connected to MySQL Server version  8.0.31-0ubuntu0.22.04.1
You're connected to database:  ('db_recruitment',)
```

4. Use systemd to manage Gunicorn

```
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo nano /etc/systemd/system/smarthire.service
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ ls
Procfile  README.md  ResumeParser.py  __pycache__  app  commands.txt  db_recruitment_file.sql  instance  my_app.py  requirements.txt  sourabh_bajaj_resume.pdf  venv  vijaysai.pdf
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ pwd
/home/ubuntu/SmartHire_CMPE272/SmartHire1
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo nano /etc/systemd/system/smarthire.service
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo nano /etc/systemd/system/smarthire.service
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo systemctl daemon-reload
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo systemctl start smarthire
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo systemctl enable smarthire
Created symlink /etc/systemd/system/multi-user.target.wants/smarthire.service → /etc/systemd/system/smarthire.service.
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ curl localhost:8000
```

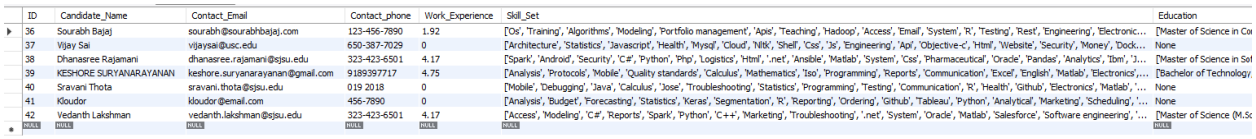
5. Run Nginx Web Server to accept and route request to Gunicorn

```
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libnginx-mod-http-geoip2 libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 nginx-common nginx-core
Suggested packages:
  fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  libnginx-mod-http-geoip2 libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2 nginx nginx-common nginx-core
0 upgraded, 9 newly installed, 0 to remove and 0 not upgraded.
Need to get 696 kB of archives.
After this operation, 2395 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nginx-common all 1.18.0-6ubuntu14.3 [40.0 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-http-geoip2 amd64 1.18.0-6ubuntu14.3 [11.9 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-http-image-filter amd64 1.18.0-6ubuntu14.3 [15.4 kB]
Get:4 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-http-xslt-filter amd64 1.18.0-6ubuntu14.3 [13.7 kB]
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-mail amd64 1.18.0-6ubuntu14.3 [45.7 kB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-stream amd64 1.18.0-6ubuntu14.3 [72.8 kB]
Get:7 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-stream-geoip2 amd64 1.18.0-6ubuntu14.3 [10.1 kB]
Get:8 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nginx-core amd64 1.18.0-6ubuntu14.3 [482 kB]
Get:9 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nginx amd64 1.18.0-6ubuntu14.3 [3882 B]
Fetched 696 kB in 0s (11.3 MB/s)
Preconfiguring packages ...
Selecting previously unselected package nginx-common.
(Reading database ... 70851 files and directories currently installed.)
Preparing to unpack .../0-nginx-common_1.18.0-6ubuntu14.3_all.deb ...
Unpacking nginx-common (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package libnginx-mod-http-geoip2.
Preparing to unpack .../1-libnginx-mod-http-geoip2_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking libnginx-mod-http-geoip2 (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package libnginx-mod-http-image-filter.
Preparing to unpack .../2-libnginx-mod-http-image-filter_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking libnginx-mod-http-image-filter (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package libnginx-mod-http-xslt-filter.
Preparing to unpack .../3-libnginx-mod-http-xslt-filter_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking libnginx-mod-http-xslt-filter (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package libnginx-mod-mail.
Preparing to unpack .../4-libnginx-mod-mail_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking libnginx-mod-mail (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package libnginx-mod-stream.
Preparing to unpack .../5-libnginx-mod-stream_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking libnginx-mod-stream (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package libnginx-mod-stream-geoip2.
Preparing to unpack .../6-libnginx-mod-stream-geoip2_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking libnginx-mod-stream-geoip2 (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package nginx-core.
Preparing to unpack .../7-nginx-core_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking nginx-core (1.18.0-6ubuntu14.3) ...
Selecting previously unselected package nginx.
Preparing to unpack .../8-nginx_1.18.0-6ubuntu14.3_amd64.deb ...
Unpacking nginx (1.18.0-6ubuntu14.3) ...
Setting up nginx-common (1.18.0-6ubuntu14.3) ...
```

```
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo systemctl start nginx
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nginx
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo nano /etc/nginx/sites-available/default
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ sudo systemctl restart nginx
(venv) ubuntu@ip-172-31-0-10:~/SmartHire_CMPE272/SmartHire1$ Connection to ec2-18-144-166-103.us-west-1.compute.amazonaws.com closed by remote host.
Connection to ec2-18-144-166-103.us-west-1.compute.amazonaws.com closed.
```

Screenshot of MySQL tables

The table Tbl_candidate_resume contains details of candidates which are taken from Amazon s3, and parsed using python parser and inserted into the table



ID	Candidate_Name	Contact_Email	Contact_phone	Work_Experience	Skill_Set	Education
36	Sourabh Bajaj	sourabh@sourabhbajaj.com	123-456-7890	1.92	['Os', 'Training', 'Algorithms', 'Modeling', 'Portfolio management', 'Apis', 'Teaching', 'Hadoop', 'Access', 'Email', 'System', 'R', 'Testing', 'Rest', 'Engineering', 'Electronic…	[Master of Science in Co
37	Vijay Sai	vijaysai@usc.edu	650-387-7029	0	['Architecture', 'Statistics', 'Javascript', 'Health', 'Mysql', 'Cloud', 'Nltk', 'Shell', 'Css', 'Js', 'Engineering', 'Api', 'Objective-c', 'Html', 'Website', 'Security', 'Money', 'Dock…	None
38	Dhanasree Rajamani	dhanasree.rajamani@sjsu.edu	323-423-6501	4.17	['Spark', 'Android', 'Security', 'C#', 'Python', 'Php', 'Logistics', 'Html', '.net', 'Ansible', 'Matlab', 'System', 'Css', 'Pharmaceutical', 'Oracle', 'Pandas', 'Analytics', 'Ibm', 'J…	[Master of Science in Sof
39	KESHORE SURYANARAYANAN	keshore.suryanarayanan@gmail.com	9189397717	4.75	['Analysis', 'Protocols', 'Mobile', 'Quality standards', 'Calculus', 'Mathematics', 'Iso', 'Programming', 'Reports', 'Communication', 'Excel', 'English', 'Matlab', 'Electronics',…	[Bachelor of Technology/
40	Sravani Thota	sravani.thota@sjsu.edu	019 2018	0	['Mobile', 'Debugging', 'Java', 'Calculus', 'Jose', 'Troubleshooting', 'Statistics', 'Programming', 'Testing', 'Communication', 'R', 'Health', 'Github', 'Electronics', 'Matlab', '…	None
41	Kloudor	kloudor@email.com	456-7890	0	['Analysis', 'Budget', 'Forecasting', 'Statistics', 'Keras', 'Segmentation', 'R', 'Reporting', 'Ordering', 'Github', 'Tableau', 'Python', 'Analytical', 'Marketing', 'Scheduling', '…	None
42	Vedanth Lakshman	vedanth.lakshman@sjsu.edu	323-423-6501	4.17	['Access', 'Modeling', 'C#', 'Reports', 'Spark', 'Python', 'C++', 'Marketing', 'Troubleshooting', '.net', 'System', 'Oracle', 'Matlab', 'Salesforce', 'Software engineering', '…	[Master of Science (M.Sc
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Project Presentation slides

https://docs.google.com/presentation/d/1_atotPp6-5P4ZGn7f9E-tSzItw4UQhUfww4nnUeBNLY/edit?usp=sharing

Milestones

1. Set up Backend(SQL and Amazon S3) - Sep 2022
2. Front end home page - Oct 2022
3. Extracting keywords for skills from the resume, add to SQL table - Oct 2022
4. Role based access for Data security - SQL - Oct 2022
5. Feature - post open jobs score candidate based on relevance to role - Oct 2022
6. Associated UI and backend changes - Oct 2022
7. Feature - shortlist candidates - Oct 2022
8. Associated UI and backend changes - Oct 2022
9. Feature - get relevant interviewers - Nov 2022
10. Associated UI and backend changes - Nov 2022
11. Feature - Add interview feedback and view feedback- Nov 2022
12. Associated UI and backend changes - Nov 2022
13. Feature - schedule interview and close job - Nov 2022
14. Associated UI and backend changes - Nov 2022
15. End-to-end Testing - Nov 2022
16. Project Hosting and Completion - Dec 1 2022

References

https://pypi.org/project/resume-parser/
https://www.nltk.org/
https://getbootstrap.com/docs/4.1/components/navbar/

https://getbootstrap.com/docs/4.1/components/carousel/
https://dev.mysql.com/doc/
https://github.com/hamziqureshi/Python_Flask_Job_Portal
https://spacy.io/usage/spacy-101
https://readthedocs.org/projects/flask/
https://medium.com/techfront/step-by-step-visual-guide-on-deploying-a-flask-application-on-aws-ec2-8e3e8b82c4f7