**Feature – Resume screening**

Currently the Hiring-Assistant (HA) system does the screening and matching together – we want it to be 2 separate processes.

**Purpose of this process:**

* Decouple Screening and Matching actions on the system.
* Normally, during the SRF life-cycle, matching is only requested/initiated after the srf has been posted for a few pre-set numbers of days.
* This is controlled by the Hiring Manager based on many criterias like
  + The SRF has accumulated a certain number of active/prospective candidates.
  + Candidate matching may only begin after the SRF is closed to applications so that we have a closed set of candidates to review.
* Screening process can be an ongoing user-initiated and/or scheduled process during the full period SRF has been active/open to application.
* The data-view created, will allow TA users to review suspect-list candidates, incomplete applications (missing-data status), Candidate\_ experience, etc. and select them for Scoring/Candidate matching Phase.

**How this should work:**

This process would be initiated via 2 methods:

* User triggered, from HA platform taking a valid SRF id as input.
* Or A scheduled daily process will run on HA for all SRFs on ORC, which are currently Active for candidates to apply for.

**User Work-Flow (User-initiated Screening)**

1. After SRF creation on ORC, User will log into the HA platform.
2. User go to SRF-Details page and will enter the SRF ID (valid on ORC) on the screen to initiate the process.
3. HA system calls ORC get-SRF API to get specific SRF details, Add/update the details on SRF-Details table and display on the screen for user to review. (below)
   * SRF-ID
   * SRF-Title
   * SRF-Status
   * HA-Process-Status, set to ‘Initiated by User-ID’, if new. Get from SRF-Details table, if already exists.
   * ORC-Fetch-Date, set to system date-time or ORC call.
   * (other SRF fields, if requested on BRD)
4. User will click on Fetch Candidate list button. This will initiate the Screening process and show user a pop-up saying – ‘Fetching Candidates for SRF-ID. Please refresh to see status.'
   * If this is the first time, HA system will call ORC to get all candidates details for the SRF and add records on Candidate\_Screening table with the data found from the screening process.
   * If screening was done previously, HA system will call ORC and check if the Candidate-ID already exists on Candidate\_Screening table for this SRF. If not found, then
5. User will refresh the page, and it should show a table with SRF-details with HA-Process-Status set to ‘Screening in process’.
6. Once the screening process completes, the page should show the status as ‘Screened’ with a button to ‘Show Candidates’, which will take them to the Candidate-list page for that SRF with a table view of all the candidates with the details populated from screening process.
7. User can also go directly to Screening page on HA Platform. This should show list of SRFs in the system, with their specific HA-Process-Status.
   * For SRFs with status Initiated, show a button to ‘Screen-candidates’, which will call the screening process.
   * For SRFs with status ‘Screened’, show a button to ‘Show Candidates’. Which will show Candidate-list page for the SRF with the details populated from screening process.
8. This screen should also have a input field for User to put in the SRF-ID and retrieve only the details of the SRF they want to view.
9. On the Candidate-List page populated for a SRF, User can then specific actions (mentioned in the **Screening-process section** below.

**Scheduled Work-Flow (Scheduled Screening)**

1. At a pre-determined time/frequency, system will invoke the following process.
2. Fetch all SRFs from ORC in Active/open to application state and add/update SRD-Details table in HA platform with HA-Process-Status set to ‘Initiated by system’.
3. For all the SRFs added to SRF-Details table and in HA-Process-Status set to ‘Initiated by system’, call the Screening process to fetch candidate details and populate Candidate-Screening table for each SRF.
4. This data will then be available for the user to view on the HA-Platform screens.

**How the System will work.**

HA platform will call ORC for SRF details and Candidate details, per SRF-ID (if user-initiated) or all SRFs in Active/open-to-application status (if Scheduled run).

HA System will process the Json data for Each SRF-candidate retrieved from ORC and write the respective Tables on its internal database

Internal Tables to be maintained by HA-Platform:

* SRF-Details Table, stores all SRFs fetched into the system for Candidate Screening with a HA-Process-Status field which will be used to Screening Dash-board Per SRF.
* Candidate-Screening table, stores all Candidate details fetched from ORC for an SRF along with the screening data fields for user-views.

**Candidate Screening Process**

For each SRF/Candidate retrieved from ORC – HA Platform will process and populate the Candidate-Screening table which would be used on the Candidate-List page on the HA platform.

The HA system will use ORC as primary data source (get-candidate API), Or using AI inference from the Resume if available.

**Table – Candidate\_Screening** maintained in HA System.

**SRF\_ID**: retrieved from ORC – mandatory, prim\_key.

**SRF\_Status**: retrieved from ORC/ or assign a status from HA platform (Retrieved).

**SRF\_Title**: retrieved from ORC, mandatory.

**Candidate\_ID**: retrieved from ORC – prim\_key, if not available on ORC, assign one in HA system

**Candidate\_Status**: assigned Ready for matching if all relevant data available ( like resume). Else Missing\_Data, if data missing on ORC.

**Candidate\_Screening\_Date:** systemdate-time stamp when data processed.

**Candidate\_Name**: Candidate full Name, if not available on ORC – pull from Resume.

**Candidate\_Job\_Role**: pull from Resume, using AI inference.

**Candidate\_Experience**: pull from resume, using AI inference.

**Candidate\_Domain/industry**: pull from resume, using AI inference.

**Candidate\_email:** pull from ORC or resume where-ever available.

**Candidate\_ph\_no:** pull from ORC or resume, where-ever available.

**Candidate\_Resume\_available**: mark Yes, if resume available and fetched from ORC.

**Candidate\_Prim\_Skills:** pull from resume, using AI inference.

**Candidate\_Sec\_Skills**: pull from resume, using AI inference.

**Candidate\_ Suspect\_Flag**: check resume/details against suspect/block list. Populate with suspect/blocked name (like institute name, company name, etc) or NA if no match found.

**Candidate\_Source:** populate data if available on ORC, or default to ORC.

**Candidate\_Suspected\_Duplication:** if the candidate’s name, email, phone number etc are already present in the DB, for any other SRF. (or same SRF)

**User\_comments**: Blank by default. Populated by users, if required.

**# additional fields for Candidate-Matching Process.**

**Candidate\_Score:**

**Candidate-Positives:**

**Candidate-Negatives:**

**AI-Comments:**

**Score\_Breakdown:**

**Industry-match**

**Education-match**

**Skills-match**

**Skills-depth**

**Certification**

**Relevance-score**

**User-Functionalities on Candidate-List page (displayed per SRF):**

This data should be available to be viewed by the user on the HA-Platform, queried per SRF.

HA platform should provide filters on the view, to show only Candidate\_Status=’Ready for Matching’ or ‘Missing\_Data’ or Candidate\_suspect\_flag set to NA (not blocked) or not-NA (blocked).

User should be able to download this data per SRF queried, into CSV file, as per the filtered view selected.

For each candidate record (or multi-select) on the view – There must be a ‘Rerun-screening’ button; which should call the fetch\_candidate API from ORC to refresh all candidate details, screen it again and populate on Candidate\_Screening\_Data table.

For each Candidate record (or multi-select) – User should be able to update Candidate\_Status to ‘Queue for Matching’.

**End-Action:**

After completing the Screening process, system should update SRF-Details table HA\_Process\_Status to ‘Screened’.

The page should allow User to select single or multiple candidate records from the view and click on a button/action ‘Select for Matching’. This will update the Candidate-Screening table Candidate-Status to ‘Ready for Matching’.

The page should allow User to select single or multiple candidate records from the view and click on a button/action ‘Remove from Matching’. This will update the Candidate-Screening table Candidate-Status to ‘Removed from Matching’.

Allow for an optional ‘Add Comments’, to update User\_comments field on Candidates-Screening table.

Show 2 checkboxes

* ‘Match All Candidates, except those marked ‘Remove from Matching’.
* ‘Match only candidate marked ‘Ready for Matching’.

HA Platform should have a button ‘Start Candidate-Matching’ for the user to invoke the Candidate matching process for all or selected ‘Queue for matching’ status candidates, based on the checkbox selected.

This Button will be grey-out, if the SRF is already in ‘Matching in Progress’ status. Show a message to user on button hover action.

Once User clicks on Show a pop-up message with count of candidates being Matched and ask to ‘Confirm’.

Once Confirmed, system will update SRF-Details table HA-Process-Status to ‘Matching in progress’ and invoke Candidate-Matching process for the SRF.

User will see a pop-up message saying ‘Candidate-Matching Initiated’. Close pop-up on OK.

Data to be pulled while screening:

"name": Full candidate name.

- "email": Candidate email address (if available).

- "industry\_domains": Relevant industries or domains the candidate has worked in (comma-separated).

- "years\_experience": Total years of professional experience (only the number, e.g., "8").

- "primary\_skill": Core technical skills the candidate is strongest in (comma-separated).

- "secondary\_skill": Other supporting or less frequently used skills (comma-separated).

**Feature – Candidate Matching**

**How this should work:**

This process would be User-Initiated from the HA-Platform.

**User Work-Flow (User-initiated Screening)**

1. Once on the Candidate-List screen;
   1. User can view screened candidate list
   2. User can filter candidates and setup for matching.
   3. User can select check box to initiate Candidate-Matching for all or selected candidates.
   4. User then clicks on ‘Start Matching’ button, and confirms on the pop-up to start the Candidate matching process.
2. Once user confirms and sees the system prompt – ‘Candidate-Matching Initiated’, they can go to the SRF page to see the status of the SRF.
3. On completion of the Candidate-Matching process, the SRF status will update to ‘Matching Completed’, and they can click on the Show matched Candidates button on the SRF screen to go to the Matched-Candidates page.
4. On the Matched candidates page, User will see top-20% of the candidates as per their scores from the matching process.
5. User can choose to view all candidates or top-n candidates from the scored list.
6. User can download the selected list into CSV format.

**How the System will work.**

Once User initiate the Candidate Matching process, system will

* Update SRF-Details Table, HA-Process-Status to ‘Matching In Progress’.
* Update Candidate-Screening table, Candidate-Status to ‘Queue for matching’
* Initiate Candidate-Matching process for specific SRF selected.

**Candidate Matching Process**

System will look for all candidate records linked to the specific SRF-ID and Candidate-Status = ‘Queue for matching’.

For each candidate, System will call ORC to fetch latest resume for each candidate and pass them thru the AI Inference model to scoring.

The process will update the following fields on Candidate-Screening table, as per the response from the AI Inference model.

**Total\_Score:**

**Candidate-Positives:**

**Candidate-Negatives:**

**AI-Comments:**

**Score\_Breakdown:**

**Industry-match**

**Education-match**

**Skills-match**

**Skills-depth**

**Certification**

**Relevance-score**

After processing each resume – update Candidate-Status to ‘Score Assigned’

After processing all candidates linked to the SRF, update SRF-Details table HA-Process-Status to ‘Matching Completed’.

Scoring Metrics:

Full Scoring Criteria (Total Score: 100):

- Industry/Domain Match: 5% (strict) — does the candidate have direct experience in the required domain(s)?

- Educational Background: 5% — relevant degrees, universities, or specializations.

- Primary + Secondary Skill Match: 30% — required and nice-to-have skills matching.

- Years of Experience or No. of Projects in Primary Skills: 20% — hands-on experience with core skills.

- Relevant Certifications: 15% — certifications mentioned in the JD or relevant to the tech stack/industry.

- Overall Resume Relevance to JD: 25% — holistic fit for the role expectations, level, and impact.

Using the above scoring parameter – following data to be generated by the LLM Model.

- positives: provide a summary of how closely the candidate matches to the job description.

- negatives: provide a summary why the candidate may not be a good match for the job description.

- additional\_comments: provide a summary of your assessment of the candidate, his skills, skill depth, certifications and overall relevance with respect to the job description

JSON format for response

```json

{

"total\_score": 0,

"score\_breakdown": {

"industry\_domain\_match": 0,

"educational\_background": 0,

"skill\_match": 0,

"skill\_depth": 0,

"certifications": 0,

"overall\_relevance": 0

},

"positives": [],

"negatives": [],

"additional\_comments": ""

}