```
graph TD
    A[Central Database]

    subgraph Bot1 [Candidate Bot]
        B1[User: Candidate]
        B2[Bot: Registration Flow]
        B3[Candidate Data Submission]
    end

    subgraph Bot2 [HR Bot]
        C1[User: HR / Recruiter]
        C2[Bot: Registration Flow]
        C3[Job Posting & Search]
    end

    subgraph WebSystem [Web Job Portal]
        W1[Web: Candidate Login/Registration]
        W2[Web: HR Login/Registration]
        W3[Job Listings & Search]
    end

    B1 --> B2 --> B3 --> A
    C1 --> C2 --> C3 --> A
    W1 --> A
    W2 --> A
    W3 --> A

    A --> B1
    A --> C1
    A --> W1
    A --> W2
    A --> W3
```

```
Website → Central DB ← Bot (Telegram/WhatsApp/Web Chat)
```

HR Flow (Sree Bot)

Step 0 — First-time registration / greeting

Trigger: User says "Hi", "Hello", or similar, and no profile exists in DB.

Sree:

Hi! I'm Sree 👋

Are you looking for:

1️⃣ Job (for candidates)

2️⃣ Candidate (for HR/recruiter)

📌 **Logic:**

- If user chooses **Job → Candidate Flow**
- If user chooses **Candidate → HR Flow**
- Store choice in DB with profile type: candidate or hr

Step 1 — HR profile creation

If **HR selected**, Sree says:

Great! Let's create your recruiter profile so I can help you find the right candidates.

Fields collected (step-by-step):

1. **Full Name**
2. **Company Name**
3. **Official Email** (OTP verification optional)

4. **Phone Number**
5. **Role / Department** (e.g., Talent Acquisition, HR Manager)
6. **Location** (HQ or branch office)
7. **Preferred way to connect** (WhatsApp / Email / Both)

📌 **Data stored in DB → hr_profiles table**

Step 2 — Main Menu (HR options)

Once registered, HR sees:

✅ Your profile is ready!

What would you like to do today?

1️⃣ Search Candidates

2️⃣ Post a Job

3️⃣ View Saved Shortlists

4️⃣ Update My Profile

Step 3 — Search Candidates

HR chooses "Search" →

Sree:

Sure! Let's narrow your search. Please provide:

1. **Role / Job Title**
2. **Skills** (must-have)
3. **Experience Range** (e.g., 3–5 yrs)
4. **Location Preference**
5. **Salary Range**
6. **Notice Period** (Immediate / < 30 days / Flexible)

📌 **Logic:**

- Sree searches DB for matching candidates.
- Shows top 3 matches initially.
- Buttons: "Show More" | "Save Candidate" | "Contact Now"

Step 4 — Post a Job

HR chooses "Post a Job" → Sree collects:

1. Job Title
2. Skills Required
3. Experience Required
4. Location
5. Salary Offered
6. Job Description (optional)
7. Deadline to Apply

📌 Job stored in DB + visible to relevant candidates in their feed.

Step 5 — View Shortlists

- HR can view candidates saved earlier.
- Can remove, contact, or download resume.

Step 6 — Update Profile

- HR can update personal or company details anytime.

What-if Scenarios

- **User sends "Hi" again later:**
  → Sree: "Welcome back, Mahesh! Want to search candidates, post a job, or view your shortlist?"

- **HR enters vague skill like 'developer':**
  → Sree: "Could you be more specific? For example: 'Java Developer', 'React Developer', etc."
- **No candidates found:**
  → Sree: "I couldn't find exact matches, but here are near matches you might like."

HR Flow – Sree Bot
(For HR users posting jobs, searching candidates, and managing applications)

---

1. HR Initiates Interaction
Bot:
"Hello 👋, welcome back [HR Name]. How can I help you today?
1️⃣ Post a new job
2️⃣ Search candidates
3️⃣ View applicants
4️⃣ Manage interviews
5️⃣ Check candidate status"

---

2. Job Posting Flow
HR chooses: "Post a new job"

1. **Bot:** "Let's post your job. Please share the **Job Title**."
2. **HR:** "Software Engineer"
3. **Bot:** "Got it ✅. Please provide:
   - Skills required
   - Minimum & maximum experience
   - Location
   - Salary range
   - Job description"
4. **HR** provides details.
5. **Bot:** "Do you want to add:
   - Remote/Hybrid/Onsite
   - Employment type (Full-time/Contract)
   - Notice period preference?"
6. **HR:** Responds.
7. **Bot:** "Posting job... ✅ Done! Job ID: HR1234.
   Would you like to:
   - Share it with all matching candidates
   - Wait for applications
   - Both?"
8. **HR** chooses option → Bot executes.

---

3. Candidate Search Flow
HR chooses: "Search candidates"

1. **Bot:** "What role are you looking for?"
2. **HR:** "Java Developer"
3. **Bot:** "Any specific filters? Please specify:
   - Skills
   - Min & max experience
   - Location
   - Salary range
   - Notice period"
4. **HR:** Gives some or all filters.

5. **Bot:** "Searching... 🔍 Found 57 candidates.
   How do you want them sorted? (Relevance, Experience, Salary, Availability)"
6. **HR** chooses sorting option.
7. **Bot:** Shows **first 5 results** with candidate cards:
   - Name (partially masked)
   - Experience & skills
   - Current CTC & expected CTC
   - Notice period
   - Location
   - [View Resume] [Message] [Shortlist] buttons
8. **What-if Scenarios**:
   - ❌ No candidates found → Bot: "No exact matches found. Do you want me to relax some filters?"
   - 📉 Low matches → Bot: "Only 3 matches found. Suggest relaxing location and salary filters."
   - 📊 Large matches → Bot: "Too many matches (500+). Suggest adding more filters."

---

4. Viewing Applicants Flow

HR chooses: "View applicants"

1. **Bot:** "Do you want to see applicants for a specific job or all jobs?"
2. **HR:** Chooses job.
3. **Bot:** Shows list of applicants in a **status table**:
   - Name
   - Applied date
   - Resume
   - Status (New, Shortlisted, Interview Scheduled, Rejected, Hired)
4. **HR** can click **[Update Status]** to change.

---

5. Interview Scheduling Flow

HR chooses: "Manage interviews"

1. **Bot:** "Select the candidate or job for interview scheduling."
2. **HR:** Picks candidate.
3. **Bot:** "Choose date, time, mode (Online/Offline), and interviewer name."
4. **HR** provides details.
5. **Bot:** "Interview scheduled ✅. Candidate notified via email/WhatsApp."

---

6. Candidate Status Check Flow

HR chooses: "Check candidate status"

1. **Bot:** "Enter candidate name or ID."
2. **HR:** Gives details.
3. **Bot:** Shows:
   - Current status
   - Last interaction date
   - Next scheduled step

---

7. What-if Scenarios for HR

Scenario	Bot Action
HR tries to search without filters	Bot asks for at least **role** or **skills**
HR tries to post incomplete job	Bot asks for missing **mandatory fields**
HR schedules overlapping interviews	Bot warns & suggests alternate slots

Scenario	Bot Action
HR requests unavailable candidate	Bot says candidate not found & suggests similar ones
HR wants to contact candidate without shortlisting	Bot allows **Quick Contact** but logs action

---

Here's the **Candidate Flow** for "Sree" with **first-time registration**, search, apply, follow-ups, and "what-if" scenarios included.

Sree Bot – Candidate Flow

1. First-time Registration

Trigger:

User sends "Hi" (or any greeting) for the first time.

Flow:

Sree: Hi! I'm Sree 👋 Are you looking for:

1️⃣ Job (Candidate)

2️⃣ Candidate (HR)

Please type 1 or 2.

If User selects 1 (Job Search as Candidate):

User profile

Profile activity (only admin can see)

Candidate updated profile: 1, 3, 7, 15, 30, 60, 90, 120, 180+ days

Candidate active: today, yesterday 3days ago, etc

A. Personal Info

Column	Data Type	Notes
full_name	string	required
email	email	required
phone	string	required
address	string	optional
social_links	list	comma-separated (LinkedIn, GitHub, etc.)
website	string	optional
father_name	string	optional
mother_name	string	optional

B. Essential Info (for Matching)

Column	Data Type	Notes
Gov/Pvt	string	Gov/pvt/Gov&Pvt
work_type	string	full-time / part-time / remote / hybrid
Preferred Industries	list	comma-separated
Department	string	comma-separated
skills	list	comma-separated
education	string	one-line summary
experience	integer	years only (0 if none)
current_role	string	"-" or "0" if no experience
expected_role	string	required

Column	Data Type	Notes
current_ctc	float	in LPA (0 if none)
expected_ctc	float	in LPA
preferred_location	list	comma-separated cities
Notice_period	float	Number only (0 if none)
relocation	boolean	yes/no
past_role_descriptions	text	long-form role description
looking_abroad	boolean	yes/no
Dob	string	DD/MMM/YYYY
gender	string	Male/Female/other
Cast	text	optional
languages_known	list	comma-separated

C. Non-Essential Info

Column	Data Type	Notes
hobbies	list	comma-separated
marital_status	string	optional
bio	text	“about me” section

✅ Chat Handler: Final Message Flow

1. `/start` or “hi”

```
👋 Welcome to Joptech — powered by Upreak 💼
Just upload your resume OR answer a few quick questions and we’ll find job matches
for you.
You have 5 credits.

Choose one:
1️⃣ Upload Resume
2️⃣ Fill Manually
```

2. Manual Fill Questionnaire (Progressive)

Question	Field	Notes
What’s your full name?	full_name	required
Your email ID?	email	must validate as email
Mobile number?	phone	digits only
Highest Education? (e.g., B.Com, M.Tech)	education	string
Your skills? (e.g., Python, Excel, AR)	skills	comma-separated
Years of experience? (number only)	experience	0 = fresher
Current Role?	current_role	skip if experience is 0
Expected Role?	expected_role	string
Current CTC? (LPA)	current_ctc	0 if fresher
Expected CTC? (LPA)	expected_ctc	float
Preferred locations? (e.g., Bangalore, Mumbai)	preferred_location	list
Gov or Pvt sector?	work_type	Gov/pvt/Gov&Pvt
Ready to relocate?	relocation	yes/no
Preferred job type?	work_type	full-time/part-time/remote

Question	Field	Notes
Looking for abroad jobs?	looking_abroad	yes/no
Date of Birth?	string	format: DD/MMM/YYYY (e.g. 15/Jan/1995)
Marital Status?	string	optional
What's your gender?	looking_abroad	yes/no
Describe past roles/responsibilities	text	multi-line input
Languages Known?	string	optional
Preferred industries?	list	comma-separated; optional
Notice period / Availability?	string	e.g., Immediate / 15 days / 1 month

📌 Fallback message (for any no input/invalid input):

```
⚠️ I didn't catch that. Please type again or use the menu below 👇
```

---

3. After Resume Upload

```
📄 Resume received!
Parsing your details and finding job matches...
```

---

4. First Job Match Results

```
🔍 Here are your top job matches!
(3-5 links shown)

⬇️ What would you like to do next?
```

✅ Buttons:

- 🔁 Show more results (X credits left)
- 💬 Get paid help via WhatsApp
- 📄 Download My Resume (AI Formatted)
- ⭐ Give Feedback / Report Crash

---

5. Out of Credits

```
🚫 You've used all your credits for today.
You'll get 5 more tomorrow. Come back soon!
```

---

6. Returning User Flow

```
👋 Welcome back, [name]!
You have 5 credits today.

Would you like to continue where we left off?
👉 Yes / No
```

If `Yes`: continue with last session.
If `No`: start fresh (new query ID generated, old cache ignored).

✅ Internal Mechanisms to Support the Flow

📌 Session Identification

- `user_id` = Telegram/WhatsApp ID
- New QID (Query ID) = `user_id + date + intent`
- Every follow-up is mapped using session + QID
- Prevents repeated results and questions

📌 Resume Versioning

- Original resume is stored as-is in user folder
- Parsed data is stored in tabular format
- One API call/day → 25 results stored → 3–5 shown per interaction
- Flags to track `shown_results`, `remaining_results`, `credits_left`

📌 Logging & Error Handling

Type	Action
API error	Log to logger module (includes timestamp + reason)
Resume parse failure	Fallback to manual questions
AI error (rate limit etc)	Try next available AI key/module
User feedback/crash	Logged with QID + session data

✅ With this foundation, you're ready to begin individual module builds:

- `chat_handler`: use this full structure
- `data_collector`: stores data per schema above
- `brain`: handles AI API logic + result processing
- `resume_builder`: uses data to auto-generate a clean, branded resume
- `credit_manager`: tracks & updates daily credits
- `feedback_center`: receives + logs user feedback

✅ Once completed → store profile in DB with unique Candidate ID.

Sree: Profile created successfully ✅ You can now search for jobs or get daily matches.
Type "search" to start searching or "my profile" to update your details anytime.

---

2. Job Search Flow

Trigger: User types "search", "find jobs", "job search", etc.

Sree: Sure! Please tell me more about the type of job you're looking for:

1️⃣ Job Title / Role?

2️⃣ Skills?

3️⃣ Experience (years)?

4️⃣ Preferred Location?

5️⃣ Expected CTC?

6️⃣ Notic1e Period?

(You can type 'skip' for any field)

📌 Filters are applied → search results fetched.

Sree: Here are 3 job matches for you 🔍 (from 20+ I found today):

[Job 1 link/details]

[Job 2 link/details]

[Job 3 link/details]

🔁 Show more results (4 credits left) | 📄 Download Resume (AI-Formatted)

---

3. Apply Flow

When Candidate clicks "Apply" or types "apply"

Sree: Applying for [Job Title] at [Company]...

✅ Application sent! You will be notified if HR views your profile or shortlists you.

---

4. Follow-up & Notifications

- **Daily job matches** at 9 AM based on saved filters.
- **Application status updates**:

Sree: Update! HR at [Company] has viewed your application.

- **Profile improvement tips** if no shortlists after X days.

---

5. What-If Scenarios

- **User has no resume uploaded yet** → Sree asks for basic details & offers AI-generated resume option.
- **User provides vague search like "IT job"** → Sree asks for more filters.
- **User's search returns 0 matches** → Sree suggests related roles or wider locations.

- **User inactive after profile creation** → Sree sends reminder:

Sree: Hey [Name], I found 12 jobs matching your profile today! Want to see them? (Yes/No)

Web dashboard flow

“ Module 2: Project (JD) Management

- **Objective:** Create detailed and specific job descriptions with clear, machine-readable requirements.
- **Main Screen (viewed within a client's page):**
  - **Header:** a back button, and a primary button **[+ Create New Project]**.
  - **Content:** A list of "Project Cards" for that client. Each card displays the Project Title, Status (e.g., Sourcing, Closed), and the Recruiter it's assigned to.
- **Project Creation/Edit Screen (/projects/new or /projects/edit/{id}):**
  - **Layout:** A structured form entered around the **Non-Negotiable Star (/) system**.
  - **The Star System:** Every key criterion has a star icon next to it.
    - **(Grey Star):** Default. Means the criterion is "preferred" or "negotiable." Used for scoring and ranking.
    - **(Yellow Star):** Activated by a click. Means the criterion is "mandatory" or "non-negotiable." Used for hard filtering.
  - **Action Buttons: Change to [Save as Draft] (Primary) and [Save & Assign] (Secondary).**
  - **Form Structure:**
    - Position / Role Name: (Text Input) [Required]
    - Number of Vacancies: (Number Input) [Required]
    - Location(s): (Repeatable Block)
    - (Text Input for Location) [Required] **[ ]**
      - **[+ Add Another Location]**
    - Qualification: (Text Input) [Optional] **[ ]**
    - CTC Range: (Text Input, e.g., "10-15 LPA") [Optional] **[ ]**
    - Notice Period: (Dropdown: "Immediate", "15, 30, 60,90 Days", etc.) [Optional] **[ ]**
    - Skills: (Tag Input Field) [Required]
      - **[+ Add Another skill]**
      - User types a skill (e.g., "Python") and hits Enter. It appears as a tag: [Python]. The user can then click the star to make it mandatory: [Python].
    - Clicking this button triggers a confirmation modal.

- A success toast is displayed after the candidate's follow-up status is successfully updated.

- **"What If" Scenarios & Solutions:**
  - **What if the criteria are too strict (e.g., PhD in AI, 15 YOE, 10 LPA CTC)?**
    - **Solution:** The system will correctly return zero matches. This is a feature, not a bug. It provides immediate, data-driven feedback to the recruiter that the criteria are unrealistic, prompting a strategic conversation with the client.
  - **What if a salesperson stars a very broad location like "India"?**
    - **Solution:** The system accepts it. The platform's job is to execute the user's instructions perfectly, giving your team the flexibility to handle any client request, from hyper-local to nationwide.

Module 3: The Recruiter's Workspace (The Engine Room)

- **Objective:** A unified command center that blends AI automation with critical human oversight. This is the most dynamic page in the application.
- **Main Screen (/clients):**
  - A list of all their assigned "Project Cards." o **Header:** A title "Projects" and a prominent button **[search]**.
  - **Content:** A list of "project Cards." Each card displays the Client Name, Spoc primary Contact Person, and the Number of candidates joined. Clicking a card takes you to that "command center" of the project.
  - ★ **Card Actions: Project Status Management (projects.html)**
  - Each project card on the projects.html page now includes a dedicated section in its footer with:
    - A **Project Status** dropdown (WIP, Hold, Closed, Win, partial win etc.). The initial value is synced with the status badge.
    - A **Remarks** text input.
    - An **"Update Status"** button.
  - Clicking "Update Status" now triggers a confirmation modal to prevent accidental changes.
  - Upon confirmation, a success notification appears, confirming that the project status has been updated.
- **Layout:** A two-column "command center" view.
  - **Left Column: "My Action Queue" (Manual Intervention)**
    - A real-time, prioritized "to-do list" of tasks the AI cannot handle alone. When a task is done, its card vanishes.
    - **Types of "Action Cards":**

- **New Matches:** "Review 5 new AI-matched candidates for [Project Name]."
- **Chat Follow-Up:** "Review chatbot conversation with [Candidate Name] - reply not understood."
- **No Response:** "Manual follow-up needed: [Candidate Name] has not responded to the bot."
- **Parse Failure:** "1 resume failed to parse. Manual data entry required."
- **New Action Card Type:** Add a new card for when the bot needs help: "[Candidate Name] requires chat intervention for [Project Name]."

- **Right Column: "Project Hub" (Automation in Action)**
  - This area dynamically changes based on what the recruiter clicks.
  - **Project Deep Dive View (when a project is selected):**

1. **Pinned Header:** Shows Project Title and the list of **Non-Negotiable** criteria so they are always visible.
2. **Core Tools (Buttons):**
   a. **[+ Upload & Parse Resumes]:** Open folder To upload new candidates profiles to this project.
   b. **[Manual Search & Add]:**
      - **Action:** Clicking this opens a pop-up/modal window.

- **Modal UI: The modal contains a search form with filters** corresponding to the main candidate portfolio fields.
  - Position / Role Name: (Text Input) [Required]
  - Location(s): (Repeatable Block)
    - (Text Input for Location) [Required]
    - [+ Add Another Location]
  - Qualification: (Text Input)
  - CTC Range: (Text Input, e.g., "10-15 LPA")
  - Notice Period: (Dropdown: "Immediate", "15 Days", “30days”, ”90days”, “60days” etc.)
  - Skills: (Tag Input Field) [Required]
  - (Text Input for Skill) [Required]
  - [+ Add Another Location]
    - **Workflow:** The recruiter fills in the search criteria and clicks **[Search Database].** A list of matching candidates from the main database appears below within the modal. They can select candidates via checkboxes and click **[Add Selected to Project]**.
      - Similar to project updates, clicking this button triggers a confirmation modal.
      - A success toast is displayed after the candidate's follow-up status is successfully updated.

Candidate List: The main area, showing a list of all candidates associated with this project. Each "Candidate Card" shows:

- Main Action Button: Above the list, there must be a [Submit Selected to Client]
- Bulk Actions: Each "Candidate Card" must have a checkbox on its left side.
- Candidate Follow-up Management:

  Each candidate card in the Recruiter Workspace has been enhanced with a new follow-up control panel. This panel includes:

  ○ A **"Follow-up Status"** dropdown (Shortlisted, Int-scheduled, offered, Joined, No show, under follow up etc.).
  ○ A **"Next Follow-up Date"** selector. ○ A **"Remarks"** text box for notes. ○ An **"Update"** button.

  - Similar to project updates, clicking this button triggers a confirmation modal.
  - A success toast is displayed after the candidate's follow-up status is successfully updated.

- Name, AI Match Score, Status Tag (New, Contacted, Replied).
- ★ **Dynamic Status Tag:** This tag is the primary indicator of chat progress. It will show states like: New, Contacting..., Awaiting Reply, Live Chat ◍, Intervention Needed ◍, Completed ✔, Declined ✖.
- ★ **Dynamic Action Buttons:** The buttons on the card change based on the status.
  - If status is New: Shows **[Initiate Chatbot].**
  - If status is Live Chat ◍ or Intervention Needed ◍: Shows **[Open Co-Pilot].**
  - If status is Completed ✔ or Declined ✖: The chat buttons are hidden.
- All states have a persistent [View Portfolio] button.
- **Action Buttons on Card:** [View/Edit Portfolio], [Initiate Chatbot], [Reject].
- **New Component:** The "Live Chat Co-Pilot" Modal
- **Trigger:** Clicking the **[Open Co-Pilot]** button on a Candidate Card.
- **Layout:** A three-pane modal view that opens on top of the workspace.
  - **Left Pane (Context):** Pinned Candidate Name, Photo, and Project Role. ○ **Centre Pane (Transcript):** The

scrollable, real-time conversation history, with distinct styling for Candidate (grey), Bot (blue), and Recruiter (green) messages.

- **Right Pane (Briefing):** Pinned list of the project's **Non-Negotiable** criteria.

- **Control Panel (at the bottom of the Centre Pane):**
  - **Default State:** A prominent **[Intervene]** button is visible. The text input field is disabled.
  - **Manual State:** After clicking **[Intervene],** the button changes to **[Resume Automation]** and the text input field becomes active.

- **"What If" Scenarios & Solutions:**
  - **What if the AI matching is good but not perfect?**
    - **Solution:** The recruiter has override power. On the Candidate Card, they can manually adjust the AI score or add a "Recruiter Approved" tag to boost a candidate's visibility.
  - **What if the Action Queue becomes overwhelming?**
    - **Solution:** The queue will have filters at the top: [Filter by Project] and [Filter by Action Type] (e.g., show only "Chat Follow-Ups"). This allows for focused work.

Module 4: The Unified Candidate Portfolio

Objective: To create a single, structured, and human-verified source of truth for every candidate, moving away from messy, unstructured resume files.

Layout: **A two-column verification and editing screen that is now** tabbed**.**

- Tabbed Interface: **The main area will have two tabs:** [Candidate Details] **and** [Full Chat History]**.**
- **Tab 1: [Candidate Details]**
  - This tab contains the full two-column layout as previously designed: the Left Column with the raw resume, and the Right Column with the editable portfolio form.
  - **Left Column (Fixed):** Displays the raw, original uploaded resume (PDF/DOCX) for constant reference.
  - **Right Column (Scrollable Form):** The editable portfolio. The AI's parsed data is pre-filled. Fields the AI is uncertain about are highlighted with a yellow border to draw attention.
- **Tab 2: [Full Chat History]**
  - **Content:** This tab displays the complete, read-only transcript of *all* conversations with that candidate across *all* projects.

- **Styling:** It uses the same distinct, color-coded bubble format (grey, blue, green) as the Live Chat Co-Pilot for consistency and clarity.
- **Functionality:** This view is for archival and review purposes. There are no active controls here.

- **Action Buttons: [Verify & Save Profile] (Primary), [Save as Draft], [Reject & Delete Profile].**

Confirmation on Delete: Clicking **[Reject & Delete Profile]** must trigger a Confirmation Modal before the action is finalized.

- **New Component 3: The Unified Candidate Portfolio Screen (Detailed UI Spec)**

This is the most important addition. We have the *list* of fields, but we need the detailed UI blueprint for the page itself, as per our established two-column design.

- **Objective:** To provide an efficient interface for recruiters to view, verify, and edit AIparsed candidate data against the original resume.
- **Layout:** A full-screen, two-column view.
  - Left Column (Fixed): A full, scrollable view of the original uploaded resume (PDF/DOCX). It is a static reference.
  - Right Column (Scrollable Form): The complete, editable portfolio form.

- Action Buttons (at the bottom of the Right Column): [Verify & Save Profile] (Primary), [Save as Draft], [Reject & Delete Profile] (with confirmation modal).
- **Section: Identity Basics** ○ Full Name: (Text Input) ○ Email Address: (Email Input) ○ Mobile Number: (Phone Input) • **Section: Education & Skills**
  - Highest Education: (Text Input) ○ Second Highest Education: (Text Input)
  - Skills: (Tag Input Field. As recruiter types, suggestions can appear. Hitting Enter creates a tag.)
  - Certificates: (Tag Input Field. Works the same as skills.)
- **Section: Job Preferences**
  - Total Experience (Years): (Number Input) ○ Current Role: (Text Input) ○ Expected Role: (Text Input)
  - Job Type: (Dropdown: Full-time, Part-time, Contract, Remote, Hybrid) ○ Current Locations: (Tag Input Field.) ○ Preferred Locations: (Tag Input Field.)
  - Ready to Relocate: (Dropdown: Yes, No, Open to Discussion)
  - Notice Period: (Dropdown: Immediate, 15 Days, 30 Days, 45 Days, 60 Days, 90+ Days)
- **Section: Salary Info**

- Current CTC (LPA): (Text Input with a number mask) ○ Expected CTC (LPA): (Text Input with a number mask) • **Section: Broader Preferences & Personal Details**
- Looking for Jobs Abroad: (Dropdown: Yes, No) ○ Sector Type: (Dropdown: Government, Private, Both)
- Preferred Industries: (Tag Input Field.)
- Gender: (Dropdown: Male, Female, Other, Prefer not to say) ○ Marital Status: (Dropdown: Single, Married, Other) ○ Date of Birth: (Date Picker component) ○ Languages Known: (Tag Input Field.) ○ Reservation Category: (Dropdown: General, OBC, SC, ST, EWS, Other)

- **Section: Work History (The Repeatable Block)**
- **UI: This section will be a container holding multiple "Job Cards."**
- **Each "Job Card" will contain:**
  - Form fields for
    - Job Title/Role,
    - Name of Company, Start Date (Date Picker), End Date (Date Picker).
    - A text area for Candidate's Key Responsibilities. A text input for Tools Used.
  - In the top-right corner of the card: small [Edit] and [Remove] icon buttons.
  - **At the bottom of the entire section: A prominent [+ Add Past Role] button, which uses JavaScript to dynamically create a new, empty "Job Card."**
- **Section: Contact and Availability (Optional)** ○ Has Current Offers?: (Dropdown: Yes, No) ○ Best Time to Contact: (Text Input, e.g., "Weekdays 4-6 PM") ○ Preferred Mode of Contact: (Dropdown: Call, WhatsApp, Email)
- **"What If" Scenarios & Solutions:**
  - **What if a candidate's updated resume is uploaded again?**
    - **Solution:** Before creating a new profile, the system checks for a duplicate Email or Phone number. If a match is found, it creates an Action Card: "Duplicate found for [Candidate Name]. Review and merge?" The verification screen then shows old and new data side-by-side for an easy merge.
  - **What if the AI completely misunderstands a resume?**
    - **Solution:** This is precisely why the human verification step is mandatory. The recruiter simply corrects the data in the form on the right while looking at the original resume on the left. The UI is the safeguard against AI error.

Module 5: The Unified Dashboard

- **Objective:** To provide a single, dynamic dashboard template that shows role-specific data, reducing development time and ensuring consistency.

- **Layout:** A single-page view of widgets.
- **2 funnel graphs based on "project status" and "candidate status"**
- **Top Bar (The Dynamic Engine):**
  - Viewing data for: [Dropdown Menu]
  - [Date Range Selector]
  - **Logic:** For Admins/Managers, the dropdown is enabled and lists their team members. For Sales/Recruiters, it's locked to their own name.
- **Widgets (Displayed as clean "cards"):**
  - **KPI Scorecards (Big Numbers):**
    - *Recruiter View:* "Candidates Submitted," "Active Projects."
    - *Sales View:* "New Clients Signed," "Projects Created."
  - **Target vs. Achievement:** A progress bar showing a key metric (e.g., "Monthly Placements: [2 / 5]").
  - **Active Project Funnel:** A bar chart showing project statuses (Sourcing, Interview, On Hold).
  - **Recent Activity Feed:** A scrolling list of key events relevant to the selected user.
  - ★ **Actionable Widgets:** All relevant chart elements and lists should be clickable. For example, clicking the "Interview Stage" bar on the Active Project Funnel should link to a pre-filtered list of those projects.
- **"What If" Scenarios & Solutions:**
  - **What if a Manager should only see their direct reports?**
    - **Solution:** The database will have a manager_id field for each user. The backend logic will only populate the dropdown with users that report to the logged-in manager.
  - **What if no performance target is set for a user?**
    - **Solution:** The Target widget will gracefully display a message: "No target has been set for this period." This prompts the user to request one. "