

MERN Engineer – Round 2 Assignment

Hybrid Application Tracking System

Problem Statement

Many application tracking systems provide automated updates via APIs, but not all applications are supported. For example, in an **interview application system**:

- **Technical role applications** have automated tracking via APIs.
- **Non-technical role applications** do not have automated tracking.

The completed solution system must:

1. **User Login:** Enable users to submit interview applications for different roles, and track status, updates, and comments.
2. **Bot Mimic Login:** Simulate automated tracking for applications with automation templates (technical roles).
3. **Admin Login:** Create Job Postings, Manage applications without automation (non-technical roles), manually updating statuses and adding comments.
4. **Full Traceability:** Maintain transparent, traceable updates with complete history for all applications.

Common Guidelines

- **Framework:** MERN Stack (MongoDB, Express.js, React, Node.js)
- **Expectations:**
 - UI / Charts library of your choice
 - Mobile-first responsive layout
 - Modular, reusable frontend components and backend APIs.
 - Clear inline comments and structured folder organization.
- **Documentation:**
 - API documentation in Markdown, along with Postman collection or Swagger UI

Submission

Submit the below in this form before 9:00 AM on 17/10/25 -
<https://zfrmz.in/ssz38H0go7VgUBvT1ezc>

- **GitHub Repository:** Include README.md with setup instructions
- **Demo Hosted:** Deploy using cloud services like Vercel or Netlify
- **API Documentation:** Postman collection or Swagger UI



Features to Implement

1. Role-Based Authentication & Dashboards

- **Authentication:** Username/Email + password using JWT.
- Dashboard for all roles to show insights, using charts, cards, stats, etc.
- **Roles:**
 - **Applicant** – Can create and track own applications.
 - **Bot Mimic** – Performs automated updates for technical roles only.
 - **Admin** – Manages non-technical applications manually & Create & manage roles for accepting applications
- **Access Control:** Each role can only see and act on applications relevant to them.

2. Application Creation & Tracking

- **Functionality:**
 - Can create new applications for any role.
 - Can view status, update history, and all notes/comments.
- **Traceability:** All application actions (manual or automated) are logged with timestamp and role (comments if passed during update).

3. Bot Mimic Login (Automated Updates for Technical Roles)

- **Purpose:** Simulates human-like updates for technical role applications with API-based workflows so that it will reflect in frontend and show like a automated system.
- **Functionality:**
 - Can progresses applications through workflow stages (e.g., "Applied → Reviewed → Interview → Offer").
 - Generates timestamped activity logs marked **Bot Mimic**.
 - Adds notes/comments to each status change.
 - Operates **only on technical role applications**; ignores non-technical roles.
 - Triggerable **on-demand** or at **scheduled intervals**.

4. Admin Manual Updates

- **Purpose:** Manage applications without automated tracking (non-technical roles).
- **Functionality:**
 - Create roles to accept applications
 - Change application statuses manually.
 - Add comments or notes for each update.
 - Ensure accurate progress tracking and audit trail for each application.
 - Dashboard to show metrics

Evaluation Criteria

- **Code Quality & Architecture:** Clean, modular, and scalable structure.
- **Functionality:** End-to-end workflow with both Bot Mimic and Admin roles working correctly.
- **UI/UX:** Intuitive dashboards, clear status timelines, responsive design.
- **Data Modeling:** Efficient schema for applications, users, and activity logs.
- **Workflow Integrity:** Logical status transitions, traceable history, and clear role separation.
- **Documentation & Version Control:** Clear README, inline comments

Deliverables

- Fully functional deployed web application.
- GitHub repository with source code and commit history.
- README .md with architecture overview, setup, and workflow instructions.
- Swagger UI (Link) or Postman collection (Exported as JSON)
- Sample user credentials for all three roles (Applicant, Bot Mimic, Admin).