Project Report – AI-Powered Proctoring Interview System

1. Introduction

This project implements a complete **AI-driven remote interview and proctoring system**. The platform enables live interviews while ensuring integrity through real-time monitoring, suspicious activity detection, and automated reporting. Both frontend and backend have been integrated to provide seamless live streaming, recording, and logging of candidate behavior.

---

2. Frontend – Interview Screen

- Developed a **web-based interface** where the interviewer can see the candidate's live video feed.
- **Two-way communication** enabled with video, audio, and chat features.
- The candidate's video is **recorded and stored securely** for post-interview review.
- Implemented **real-time overlays** to indicate recording status and proctoring alerts.
- Interviewer dashboard displays **live suspicious events** (e.g., "Candidate looking away", "Multiple faces detected").

---

3. Focus Detection Logic

To ensure the candidate's attentiveness, we implemented advanced focus detection using **OpenCV, MediaPipe, and TensorFlow.js**.

- **Face Tracking:** Detects presence and position of the candidate's face in real-time.
- **Looking Away Detection:** Flags and logs if the candidate is not looking at the screen for more than **5 seconds**.
- **Face Absence:** Logs an event if no face is present for over **10 seconds**.
- **Multiple Face Detection:** Identifies and logs if more than one person is present in the frame, indicating possible malpractice.
- **Event Logging:** All focus-related events are logged with timestamps and candidate identity for reporting.

---

4. Suspicious Item/Note Detection

We integrated object detection models (**YOLOv5 + TensorFlow.js**) to detect unauthorized items in the candidate's environment.

- **Mobile Phone Detection:** Detects if a mobile phone is visible in the frame.

- **Books/Paper Notes Detection:** Identifies presence of study materials or written notes.
- **Extra Electronic Devices:** Recognizes additional electronic items such as calculators, tablets, or secondary screens.
- All such detections are **flagged in real-time** and logged with timestamps.
- Visual alerts are shown to the interviewer, ensuring transparency.

---

5. Backend Integration

The backend was implemented to manage data storage and reporting.

- **Database:** Logs are stored in a database (**MongoDB**). Each entry contains:
    - Candidate Name
    - Session ID
    - Timestamp
    - Event Type (Focus Lost, Multiple Faces, Phone Detected, etc.)
- **API Services:** REST APIs are developed to:
    - Fetch candidate activity logs
    - Retrieve suspicious event reports
    - Access video recordings
- **Scalable Design:** Backend can handle multiple concurrent interview sessions.

---

6. Reporting and Analytics

At the end of every session, the system automatically generates a **Proctoring Report**.

Report Includes:

- **Candidate Details:** Name, Session ID, and Interview Duration.
- **Focus Analysis:** Number of times the candidate looked away (>5 sec), absence (>10 sec), or multiple faces detected.
- **Suspicious Events:** Frequency of mobile phone, notes, or extra devices detected.
- **Timestamps:** Every suspicious event is recorded with exact timestamps.
- **Final Integrity Score:**
    - Starts from 100

  - Deductions applied for each suspicious activity
  - Final score represents candidate’s integrity during the interview

Example:

- Candidate Name: John Doe
- Duration: 42 minutes
- Focus Lost: 3 times
- No Face Detected: 2 times
- Multiple Faces: 1 instance
- Phone Detected: 1 instance
- **Final Integrity Score: 82/100**

---

7. Key Features Implemented

- Live Video & Audio Streaming (Two-way)
- Candidate Video Recording & Secure Storage
- Real-Time Chat between Interviewer & Candidate
- Focus Detection (Looking Away, No Face, Multiple Faces)
- Suspicious Object Detection (Phones, Notes, Devices)
- Real-Time Alerts for Interviewer
- Automatic Event Logging with Timestamps
- Backend API & Database Storage (MongoDB)
- Automated Proctoring Report Generation with Integrity Score

---

8. Conclusion

The project successfully delivers a **fully functional AI-powered proctoring system** tailored for remote interviews. It enhances transparency, ensures fairness, and provides actionable insights to interviewers. With real-time monitoring, detailed logging, and automated reporting, this system can be directly deployed in real-world recruitment or academic proctoring scenarios.