

Risk-Based Proctoring System

– Technical Architecture & Workflow

TEAM:CODECRAFTERS

- MEGHANA M
- PREETHI M S
- RAKSHITHA D V
- SHAMBHAVI MP

TRACK-01

Problem we are solving:

Traditional proctoring methods rely on intrusive video surveillance, raising privacy concerns and causing discomfort for test-takers. There is a need for a non-invasive, AI-powered solution that ensures academic integrity without violating privacy.

Why it is an important problem??

Webcam-based proctoring compromises privacy, elevates stress, and negatively impacts performance. AI-driven facial recognition is prone to biases, leading to false accusations. The need for controlled environments creates disparities for students lacking private spaces. Growing legal challenges and student opposition highlight the urgent demand for a more ethical and advanced solution.

Primary users affected by this issue:

- **Students:**

Experience stress, privacy invasion, and potential false accusations.

- **Educational Institutions:**

Face legal challenges, student dissatisfaction, and reputational risks.

- **Examiners & Proctors:**

Require reliable yet non-invasive solutions to uphold exam integrity.

Solution Overview

Solution Description:

Our AI-powered proctoring system eliminates intrusive webcam surveillance by leveraging non-behavioral movements such as keystroke dynamics, mouse movements, inactive time, copy-paste actions, long holds, and tab switches. Additionally, LSTM-based AI models analyze mouse gaze and cursor dynamics using sample data for training. Remote access breaches are detected via IP monitoring and USB device tracking, ensuring a secure exam environment.

How It Effectively Solves the Problem:

- Privacy-First Approach:

Eliminates the need for video surveillance, addressing privacy concerns.

- **Real-Time Scalable Risk Detection:**

Uses a risk engine (Low, Medium, High) to assess suspicious behavior.

- **Works in Low Bandwidth Conditions:**

Unlike video-based solutions, it operates efficiently even with limited connectivity.

- **External Device & IP Monitoring :**

Detects unauthorized access attempts and external hardware connections.

Unique Differentiation:

- **Non-Intrusive Behavioral Monitoring:**

Focuses on digital activity rather than physical surveillance.

- **Scalability & Cost-Effectiveness:**

AI-driven approach reduces manual proctoring costs while scaling effortlessly.

- **Advanced AI Algorithms:**

LSTM models enhance accuracy in detecting anomalies.

- **Remote Access Prevention:**

Identifies and blocks unauthorized remote access attempts.

Our solution ensures secure, fair, and accessible online exams while prioritizing student privacy and institutional integrity.

System Architecture Overview

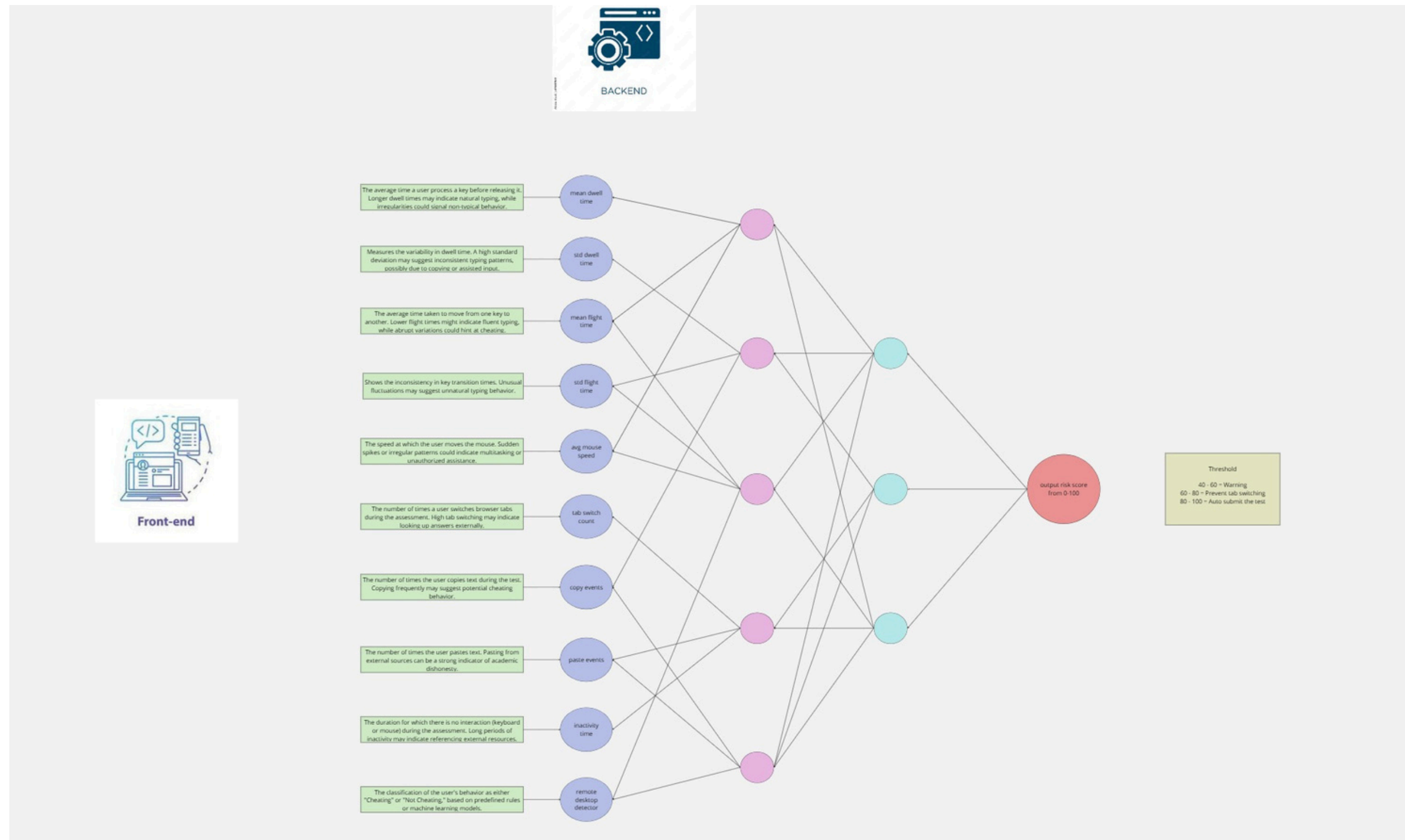


fig: Architecture of Risk-Based Proctoring System

The Risk-Based Proctoring System for Online Assessments is designed to ensure academic integrity without intrusive video surveillance. The architecture consists of multiple interconnected components working together to analyze user behavior, assess risk, and apply adaptive interventions.

Key Components of the System:

Frontend (User Interface & Monitoring):

- Developed using React.js for a responsive and interactive experience.
- Utilizes Browser APIs to capture user activity such as:
 - > Mouse movements (irregular patterns, inactivity). and copy, paste monitoring
 - > Keystroke dynamics (speed, consistency, delays).
 - > Tab switching detection (monitoring active window focus).
 - > Geolocation-based fraud detection
- Communicates with the backend in real-time using WebSockets (Socket.io).

Real-time Data Transfer (WebSockets):

- Enables continuous monitoring of user interactions without page reloads.
- Ensures low-latency communication between the frontend and backend.
- Sends behavioral data packets at regular intervals for AI processing.

Backend (Processing & API Layer):

- Built with Node.js + Express.js to handle API requests efficiently.
- Receives real-time behavioral data from the frontend via WebSockets.
- Passes data to the AI Risk Engine for analysis and scoring.
- Implements role-based access control for secure data handling.

Risk Engine (AI/ML Model for Risk Assessment):

- Developed in Python (Flask) api with lstm in TensorFlow.
- Processes incoming behavioral data to detect suspicious patterns.
- Dynamically calculates a risk score (Low, Medium, High).
- Applies adaptive interventions (e.g., warnings, restrictions) based on risk level.

Database (Storage & Logs):

- Uses MongoDB (NoSQL) for storing behavioral logs and risk scores..
- Maintains audit logs for review by proctors.


Admin Dashboard (Monitoring & Control Panel):

- Built with React.js + Node.js for administrators to monitor test sessions.
- Displays real-time risk scores, alerts, and flagged candidates.
- Provides tools for manual review, risk score adjustments, and intervention control.

Data Flow & Workflow

- User Logs in – The candidate starts the online exam via the frontend (React.js).
- Behavioral Data Captured – The system monitors keystrokes, mouse movements, and tab switches using Browser APIs.
- Real-time Data Transmission – The collected data is sent to the backend via WebSockets for continuous monitoring.
- Risk Score Calculation – The backend processes the data, and an AI-based model updates the risk score dynamically.
- Decision Making – If the risk score crosses predefined thresholds:
 - Low Risk → No action.
 - Medium Risk → Warning message displayed.
 - High Risk → Tab switching blocked or administrator notified.
- Logging & Reporting – All user actions and risk scores are stored in the database for review by exam administrators.
- Admin Review & Interventions – If necessary, proctors can manually review flagged cases via an admin dashboard.

Technologies & Tools Used

- Frontend: React.js, Browser APIs
 - Backend: Node.js, Express.js
 - Real-time Communication: WebSockets (Socket.io)
 - Machine Learning: Python (Flask, Scikit-learn, TensorFlow/PyTorch)
 - Database: MongoDB
- 

Why we ??

Our AI-driven proctoring system redefines online exam security by offering a non-intrusive, privacy-first, and highly effective fraud detection approach. Unlike traditional webcam-based proctoring, which raises privacy concerns and requires high-speed internet, our solution ensures integrity through advanced behavioral analytics and AI-driven risk assessment.

What Makes Us Stand Out?

- ✓ Privacy-First Approach – No intrusive webcam or microphone monitoring, ensuring a stress-free test-taking experience.
- ✓ AI-Powered Behavioral Monitoring – Detects anomalies using keystroke dynamics, mouse movements, inactive time, copy-paste actions, long holds, and tab switches without violating user privacy.

- ✅ LSTM-Based AI for Gaze & Cursor Tracking – Our deep learning models analyze mouse gaze and cursor behavior to detect suspicious activities, making it harder to cheat.
- ✅ Advanced IP & USB Monitoring – Detects remote access attempts, proxy/VPN usage, and unauthorized USB devices, preventing external aid.
- ✅ Works in Low Bandwidth Conditions – Unlike video proctoring, our system operates efficiently even with limited internet connectivity, making it accessible for all students.
- ✅ Scalable & Cost-Effective – AI automation reduces the need for manual proctors, making it an ideal solution for large-scale online exams while cutting down operational costs.

By leveraging cutting-edge AI, deep learning models, and real-time fraud detection mechanisms, we ensure a secure, fair, and privacy-friendly online exam experience. Our system not only upholds academic integrity but also fosters a more ethical and accessible approach to digital assessments. 🚀

THANK YOU