

---

## **Project Name:**

***Attention Keeper***

## **Project Sponsor:**

***DR. Al-Hassan Mohamed***

## **Project Manager:**

***Fatma Al-zahraa Alaa Mahmoud***

***Alaa Atef Ali Mohamed***

***Mariam Tarek Khalaf***

***Jana Alaa Al-Deen Ahmed***

## **Date:**

October 15, 2025

---

## **Project Purpose and Justification:**

The purpose of this project is to develop an **Eye-Tracking Browser Extension** that monitors a user's eye movement while studying or watching videos. The system automatically pauses the video when the user's eyes move away from the screen, and when the user looks back, it rewinds the video by five seconds before resuming playback. This project aims to enhance focus, minimize distractions, and improve learning efficiency by ensuring continuous attention. It aligns with the goal of utilizing smart automation to support effective digital learning experiences.

---

## **Project Description:**

This project involves developing a **browser extension** that uses **eye-tracking technology** to monitor user attention while studying or watching videos.

---

## **Project Objectives:**

- Develop and deploy the eye-tracking browser extension by **April 15, 2026**
  - Ensure compatibility with **all major browsers** (Chrome, Edge, Firefox, Safari)
  - Improve user focus and reduce off-screen distraction time by **at least 70%**
  - Maintain smooth video control with **less than 1% system error rate**
- 

## **Deliverables:**

- A fully functional **AI-powered eye-tracking browser extension** compatible with all major browsers (Chrome, Edge, Firefox, Safari)
  - An intuitive **user interface** for video control and eye-tracking calibration
  - A **machine learning model** for real-time eye movement detection and focus analysis
  - **Documentation** including installation guide, user manual, and technical report
  - **Testing and performance reports** demonstrating accuracy and reliability.
-

---

## **Milestones & Timeline:**

- Requirements gathering: Oct 2025
  - Design prototype: Nov 2025
  - Development: Dec 2025 – Mar 2026
  - Testing: Dec 2025 – Mar 2026
  - Deployment & training: Apr 2026
- 

## **Technical Requirements:**

• Platform:

Web-based browser extension compatible with all modern browsers, including Chrome, Firefox, Edge, Opera, Brave, and Safari. Built using JavaScript, HTML, and CSS with Web Extension API and Manifest V3 for cross-browser support.

• Operating System:

Should function properly on Windows 10/11, macOS, and Linux systems where supported browsers run.

• Hardware:

Requires a built-in or external webcam with a minimum resolution of 720p for accurate eye tracking. No external hardware or additional plugins needed.

• Artificial Intelligence and Computer Vision:

Integrates AI-based computer vision models using libraries such as TensorFlow.js, Media Pipe, or WebGazer.js to detect and analyze eye movements in real time.

• Browser and Video Integration:

Automatically detects video elements on any webpage, pauses playback when the user looks away, and resumes while rewinding the video 5 seconds when attention returns. Implemented using HTML5 Video API and DOM manipulation.

• Performance:

Must process real-time eye-tracking at a minimum of 15 FPS with low CPU and memory usage. Uses asynchronous JavaScript, Web Workers, and optimized algorithms for smooth performance.

• Security and Privacy:

All AI processing occurs locally on the user's device. No image or behavioral data is transmitted externally. Follows privacy-by-design principles and browser security standards (TLS, HTTPS).

• User Interface:

Simple and responsive design that displays clear status indicators (e.g., "Video Paused" or "Playback Resumed"). Developed using HTML, CSS, and modern frameworks such as Tailwind CSS or Bootstrap.

• Automation:

Automated playback control based on user attention. Includes customizable rewind duration (default 5 seconds) and intelligent timing management with JavaScript event handlers.

• Ad Blocking System:

Built-in smart ad blocker removes advertisements before, during, or around video playback without affecting legitimate content. Uses DOM analysis, CSS selectors, and Mutation Observer for real-time ad detection.

---

---

## **Limits and Exclusions:**

- This project will not develop a mobile application version of the attendance system; only a web-based platform is included.
  - External Integration: The extension will not integrate with any external streaming services or other software systems. Its functionality will be limited solely to standard video elements within the browser.
  - Hardware: The project does not include providing or funding external webcams or any additional hardware. The user is responsible for having a compatible device equipped with an appropriate camera.
  - Cloud Processing: No cloud-based AI services or online image processing will be used. All processing will occur locally within the browser on the user's device only.
  - Data Storage: The extension will not store user data, video history, or webcam snapshots on any external servers. Processing will occur temporarily in memory only during operation.
- 

## **Approval:**

- Project Sponsor: \_\_\_\_\_ Date: \_\_\_\_\_
  - Project Manager: \_\_\_\_\_ Date: \_\_\_\_\_
-