

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA INFORMÁTICA EN
TECNOLOGÍAS DE LA INFORMACIÓN



UNIVERSITAS
Miguel Hernández

“DESARROLLO DE UNA APLICACIÓN PARA
EL CONTROL DE LA ATENCIÓN”

TRABAJO DE FIN DE GRADO

Junio – 2025

AUTOR: Javier Maciá Sempere

DIRECTOR: Federico Botella Beviá

RESUMEN

La atención, junto con la percepción y la memoria, constituye un proceso psicológico básico e inherente al ser humano. En un contexto educativo cada vez más influido por las tecnologías digitales, resulta fundamental contar con herramientas que permitan analizar, de forma objetiva y automatizada, el grado de atención de los estudiantes durante una clase con el fin de estudiar cómo funcionan mejor las sesiones del profesorado. Con este propósito, en este Trabajo de Fin de Grado se ha diseñado y desarrollado una aplicación web capaz de monitorizar en tiempo real el nivel de atención y somnolencia de los alumnos mediante técnicas de inteligencia artificial y visión por ordenador.

La solución propuesta se integra como una herramienta auxiliar a las sesiones de clase tradicionales. A través de la cámara del dispositivo del estudiante, el sistema analiza diversos indicadores como la dirección de la cabeza, el parpadeo y el grado de apertura ocular, y cómo estos varían a lo largo de la sesión. A partir de estos datos, se clasifica el estado del estudiante en una de las siguientes categorías: atento-despierto, atento-somnoliento, distraído-despierto o distraído-somnoliento. Esta información se notifica al docente mediante una interfaz gráfica clara e intuitiva. Además, los datos quedan almacenados y representados por medio de gráficas para su análisis, garantizando en todo momento el anonimato del alumnado.

La aplicación ha sido desarrollada empleando tecnologías modernas como Next.js, React o MediaPipe, y está diseñada para ser multiplataforma, accesible desde cualquier navegador y dispositivo con cámara. Se ha priorizado la ligereza del sistema para garantizar un bajo consumo de recursos y una buena experiencia de usuario, tanto para estudiantes como para profesores.

Este proyecto combina conocimientos de ingeniería del software, inteligencia artificial, desarrollo web *full-stack* y gestión de bases de datos. Asimismo, plantea una solución ética y respetuosa con la privacidad, evitando el almacenamiento de imágenes o datos personales.

Los resultados obtenidos demuestran que es posible construir un sistema funcional y eficaz para el análisis de la atención en entornos educativos reales.

AGRADECIMIENTOS Y DEDICATORIAS

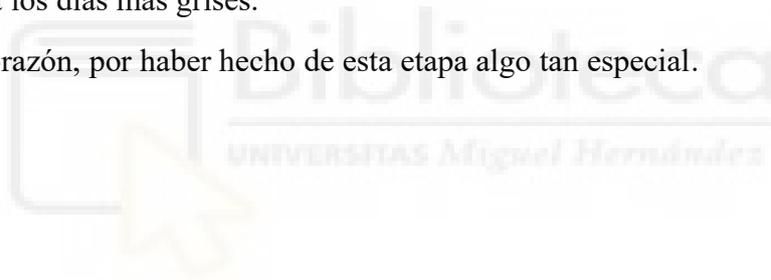
Me gustaría agradecer a mi hermano, Sergio, por haberme dado apoyo tanto apoyo emocional durante todo el desarrollo de este proyecto. Mi hermano, para mí, siempre ha sido un ejemplo a seguir y, sin quererlo, me he dado cuenta de que siempre he estado siguiendo sus mismos pasos. Gracias, tete, por siempre haberme aconsejado y advertido de todo lo que podría o no haber pasado.

Además, me gustaría agradecer enormemente a mis padres, que siempre han intentado hacer todo lo posible por mí. Se han sacrificado no solo a lo largo de estos años realizando el grado, si no también antes de él. Me han apoyado siempre, han confiado en mí y me han dado todos los recursos necesarios para que yo creciera como persona.

A lo largo de este tiempo he hecho amistades en la carrera muy bonitas que guardaré siempre conmigo, y con la que recordaremos momentos especialmente agradables. Gracias a todos esos compañeros y amigos que siempre han estado ahí apoyándome no solamente de manera emocional, sino también profesional.

Y por supuesto, gracias a Aroa, quién es de las personas que más ha podido vivir toda esta aventura. Ella siempre me ha ofrecido apoyo de cualquier tipo, ha sabido darme calma en los momentos de estrés y ha puesto siempre esa preciosa sonrisa que tanto ha animado hasta los días más grises.

Gracias, de corazón, por haber hecho de esta etapa algo tan especial.



ÍNDICE

| | |
|--|-----------|
| 1. INTRODUCCIÓN | 10 |
| 1.1. OBJETIVOS | 10 |
| 1.2. JUSTIFICACIÓN DEL PROYECTO | 11 |
| 1.3. CONTEXTO E IMPORTANCIA | 12 |
| 1.4. MOTIVACIONES PERSONALES | 12 |
| 1.5. LIMITACIONES DEL PROYECTO | 13 |
| 2. ESTADO DEL ARTE | 15 |
| 2.1. INTELIGENCIA ARTIFICIAL | 15 |
| 2.1.1. <i>Aprendizaje automático</i> | 15 |
| 2.1.2. <i>Redes neuronales y deep learning</i> | 16 |
| 2.1.3. <i>Legislación sobre el reconocimiento facial</i> | 17 |
| 2.2. INTRODUCCIÓN A LA ATENCIÓN | 17 |
| 2.2.1. <i>Técnicas de visión por computadora para la medición de la atención</i> | 18 |
| 2.2.1.1. Seguimiento de la mirada | 18 |
| 2.2.2. <i>Cómo mediremos la atención en este TFG</i> | 21 |
| 2.3. INTRODUCCIÓN AL ESTADO DE SOMNOLENCIA | 21 |
| 2.3.1. <i>Detección de características a partir del pestañeo</i> | 22 |
| 2.3.1.1. Pestañeo y el descanso mental | 22 |
| 2.3.1.2. Detección y cálculo de un parpadeo | 25 |
| 3. HIPÓTESIS DEL TRABAJO | 28 |
| 3.1. DISPOSITIVOS Y ESTRUCTURA | 29 |
| 3.1.1. <i>Características del servidor</i> | 30 |
| 3.1.2. <i>Características del cliente</i> | 30 |
| 3.2. LENGUAJES DE PROGRAMACIÓN Y LIBRERÍAS | 30 |
| 3.2.1. <i>Servicio web</i> | 31 |
| 3.2.1.1. HTML | 31 |
| 3.2.1.2. CSS | 31 |
| 3.2.1.3. Tailwind | 31 |
| 3.2.1.4. JavaScript | 32 |
| 3.2.1.5. Typescript | 32 |
| 3.2.1.6. React | 32 |
| 3.2.1.7. Next.js | 33 |
| 3.2.1.8. Node.JS | 33 |
| 3.2.1.9. MySQL | 33 |
| 3.2.1.10. SQL | 33 |
| 3.2.1.11. Socket.IO | 34 |
| 3.2.1.12. Prisma | 34 |
| 3.2.2. <i>Tecnologías para la Inteligencia Artificial</i> | 34 |
| 3.2.2.1. Mediapipe | 34 |
| 3.2.3. <i>Justificación de las tecnologías</i> | 34 |
| 4. METODOLOGÍA Y RESULTADOS | 37 |
| 4.1. CICLO DE VIDA | 37 |
| 4.1.1. <i>Metodología Kanban</i> | 37 |
| 4.2. ESPECIFICACIÓN | 38 |
| 4.2.1. <i>Requisitos del software</i> | 38 |
| 4.2.1.1. Requisitos funcionales | 39 |
| 4.2.1.2. Requisitos no funcionales | 41 |
| 4.2.2. <i>Actores</i> | 42 |
| 4.2.3. <i>Casos de uso</i> | 43 |

| | | |
|---------------|--|-----------|
| 4.3. | DISEÑO | 55 |
| 4.3.1. | <i>Diagrama de casos de uso</i> | 55 |
| 4.3.2. | <i>Diagrama de estados</i> | 56 |
| 4.3.3. | <i>Diagrama de secuencia</i> | 58 |
| 4.3.4. | <i>Diagrama de flujo</i> | 59 |
| 4.3.5. | <i>Diagrama de clase</i> | 61 |
| 4.3.6. | <i>Diagrama de base de datos</i> | 61 |
| 4.3.7. | <i>Informes de los datos capturados</i> | 63 |
| 4.3.7.1. | Datos clave a detectar..... | 63 |
| 4.3.7.2. | Consideraciones técnicas de captura y procesamiento de datos | 64 |
| 4.3.8. | <i>Diseño de la interfaz</i> | 66 |
| 4.3.8.1. | CARACTERÍSTICAS de la interfaz | 66 |
| 4.3.8.2. | Bocetos de la interfaz | 66 |
| 4.4. | ESTUDIO DE LA EVOLUCIÓN DEL PROYECTO | 70 |
| 4.4.1. | <i>Idea inicial y primeras etapas</i> | 70 |
| 4.4.2. | <i>Problemáticas enfrentadas</i> | 71 |
| 4.4.2.1. | Balance de cargas | 71 |
| 4.4.2.2. | Datos anonimizados..... | 73 |
| 4.4.2.3. | Selección de las métricas | 74 |
| 4.4.2.4. | Gestión de múltiples conexiones simultáneas..... | 74 |
| 4.5. | RESULTADOS FINALES | 75 |
| 4.5.1. | <i>Página principal</i> | 75 |
| 4.5.2. | <i>Página de registro</i> | 77 |
| 4.5.3. | <i>Panel del profesor</i> | 78 |
| 4.5.4. | <i>Clase</i> | 81 |
| 5. | CONCLUSIONES | 84 |
| 5.1. | AMPLIACIONES FUTURAS..... | 84 |
| ANEXOS | | 86 |
| | ANEXO I: MANUAL DE INSTALACIÓN | 87 |
| 6. | BIBLIOGRAFÍA | 90 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 3.1—1 Especificaciones del ordenador | 30 |
| Tabla 4.2—1 Requisito funcional: Captura de la dirección de la cabeza..... | 39 |
| Tabla 4.2—2 Requisito funcional: Detección de parpadeos y ojos cerrados | 39 |
| Tabla 4.2—3 Requisito funcional: Medición del tiempo de la atención..... | 39 |
| Tabla 4.2—4 Requisito funcional: Medición del tiempo de la somnolencia | 39 |
| Tabla 4.2—5 Requisito funcional: Grabación de las sesiones | 40 |
| Tabla 4.2—6 Requisito funcional: Sistema en tiempo real | 40 |
| Tabla 4.2—7 Requisito funcional: Administración de las sesiones..... | 40 |
| Tabla 4.2—8 Requisito funcional: Administración del perfil | 40 |
| Tabla 4.2—9 Requisito funcional: Resultados globales | 40 |
| Tabla 4.2—10 Requisito funcional: Resultados individuales..... | 40 |
| Tabla 4.2—11 Requisito no funcional: Notificación al profesor de la ausencia de atención | 41 |
| Tabla 4.2—12 Requisito no funcional: Notificación al profesor de la somnolencia | 41 |
| Tabla 4.2—13 Requisito no funcional: Diseño estético, minimalista y sencillo | 41 |
| Tabla 4.2—14 Anonimato de los alumnos | 41 |
| Tabla 4.2—15 Requisito no funcional: Código escalable | 41 |
| Tabla 4.2—16 Requisito no funcional: Código modular | 42 |
| Tabla 4.2—17 Requisito no funcional: Librerías de código abierto..... | 42 |
| Tabla 4.2—18 Actor: Profesor | 42 |
| Tabla 4.2—19 Actor: Estudiante..... | 42 |
| Tabla 4.2—20 Actor: Inteligencia Artificial (IA) | 42 |
| Tabla 4.2—21 Caso de uso 1: Visualizar panel principal | 44 |
| Tabla 4.2—22 Caso de uso 2: Iniciar la clase | 45 |
| Tabla 4.2—23 Caso de uso 3: Parar grabación..... | 46 |
| Tabla 4.2—24 Caso de uso 4: Visualizar sesión..... | 47 |
| Tabla 4.2—25 Caso de uso 5: Eliminar una sesión | 48 |
| Tabla 4.2—26 Caso de uso 6: Modificar el perfil | 48 |
| Tabla 4.2—27 Caso de uso 7: Sostener la mirada | 49 |
| Tabla 4.2—28 Caso de uso 8: Determinar una apertura de ojos y una frecuencia de parpadeo | 49 |
| Tabla 4.2—29 Caso de uso 9: Determinar nivel de atención | 50 |
| Tabla 4.2—30 Caso de uso 10: Generar informe individual | 51 |
| Tabla 4.2—31 Caso de uso 11: Generar informe global | 52 |
| Tabla 4.2—32 Caso de uso 12: Mostrar notificaciones | 53 |
| Tabla 4.2—33 Caso de uso 13: Asistir a la clase..... | 54 |
| Tabla 4.2—34 Caso de uso 14: Salir de la clase..... | 54 |
| Tabla 4.3—4 Iconografía del diagrama de flujo..... | 59 |
| Tabla 4.3—8 Columnas de la tabla de Profesor | 62 |
| Tabla 4.3—9 Columnas de la tabla de Alumno | 62 |
| Tabla 4.3—10 Columnas de la tabla de Sesión | 62 |
| Tabla 4.3—11 Columnas de la tabla de Evaluación | 63 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 2.1—1 Ejemplo de una red neuronal [6]..... | 16 |
| Figura 2.2—1 Métodos de seguimiento de la mirada | 19 |
| Figura 2.3—1 Relación de la activación cerebral con el parpadeo | 23 |
| Figura 2.3—2 Variación de indicadores relacionados con la atención y somnolencia a lo largo del día | 24 |
| Figura 2.3—3 Puntos utilizados para estimar el EAR | 26 |
| Figura 4.3—1 Diagrama de casos de uso | 56 |
| Figura 4.3—2 Diagrama de estado | 57 |
| Figura 4.3—3 Diagrama de secuencia | 58 |
| Figura 4.3—5 Diagrama de flujo..... | 60 |
| Figura 4.3—6 Diagrama de clase | 61 |
| Figura 4.3—7 Diagrama de base de datos..... | 61 |
| Figura 4.3—12 Boceto del inicio de sesión, estudiante seleccionado | 67 |
| Figura 4.3—13 Boceto del inicio de sesión, profesor seleccionado..... | 67 |
| Figura 4.3—14 Boceto de la vista de inicio del profesor..... | 68 |
| Figura 4.3—15 Boceto de la vista de las estadísticas de una clase | 69 |
| Figura 4.4—1 Cámara Intel Realsense D435 | 70 |
| Figura 4.4—2 Problemática de una lógica centrada en el servidor | 72 |
| Figura 4.4—3 Optimización con la lógica centrada en el cliente..... | 73 |
| Figura 4.5—1 Página principal del sitio web | 76 |
| Figura 4.5—2 Acceso como profesor | 77 |
| Figura 4.5—3 Página de registro | 77 |
| Figura 4.5—4 Página principal del acceso identificado | 78 |
| Figura 4.5—5 Resultados de la atención de una clase..... | 79 |
| Figura 4.5—6 Cambiar alias del profesor | 80 |
| Figura 4.5—7 Cambiar contraseña del profesor..... | 80 |
| Figura 4.5—8 Eliminar cuenta del profesor | 81 |
| Figura 4.5—9 Interfaz de la clase del profesor | 81 |
| Figura 4.5—10 Interfaz del alumno ante signos de somnolencia..... | 82 |
| Figura 4.5—11 Interfaz del usuario ante cambios en la atención..... | 82 |
| Figura 0—1 Contenido del archivo .env | 87 |
| Figura 0—2 Crear la base de datos..... | 88 |

Capítulo 1

Introducción



Biblioteca
UNIVERSITAS

1. INTRODUCCIÓN

En la actualidad, la educación ha evolucionado en gran medida gracias a las Tecnologías de la Información y Comunicación (TIC). Los estudiantes son capaces de aprender gracias a estas tecnologías, ya que permiten estudiar desde cualquier lugar si se tiene acceso a Internet. Una buena educación ciudadana es fundamental para el correcto desarrollo de una sociedad, y es por ese motivo que la mayoría de los países desarrollados ponen sus esfuerzos en estas tecnologías [1].

Es importante y resulta de interés enseñar de maneras innovadoras y eficientes, pero para ello, a veces es necesario entender y estudiar lo básico que hace que funcione el proceso de aprendizaje: ¿qué es aquello que capta y mantiene la atención de los alumnos? ¿Dónde se reparte el foco de los alumnos durante una clase? En este trabajo, se desarrolla una solución software que permite el análisis y seguimiento de la atención en un aula virtual en tiempo real mediante técnicas de inteligencia artificial, persiguiendo el fin de proporcionar a los organismos educativos una solución para potenciar más el estudio del desempeño de sus clases.

1.1. OBJETIVOS

El propósito de este proyecto es desarrollar una aplicación web para monitorizar la atención de unos usuarios en tiempo real durante las sesiones de clase. El enfoque de esta idea es que el programa realice una función complementaria al sistema empleado para la impartición de las clases del profesorado de un centro educativo. Los alumnos asistirán de manera presencial a las clases y activarán el software para que su atención sea analizada de manera anónima.

Este estudio de la atención se realiza capturando imágenes de las cámaras de los diferentes estudiantes que componen la clase en directo. Los resultados serán presentados en tiempo real al responsable de dirigir la clase por medio de una interfaz de usuario intuitiva. Asimismo, estos resultados también se almacenarán en una base de datos para su posterior visualización por parte del profesor, tanto de manera global como individual.

Para determinar si el estudiante está prestando suficiente atención o no, se analiza la duración en la que el usuario mantiene la mirada en un mismo punto, así como si muestra signos de somnolencia al pestañear en exceso o tener muy cerrados los ojos. Haciendo uso de técnicas de visión por ordenador e inteligencia artificial, se puede obtener información de la atención y diferenciar entre cuatro estados de atención diferentes: atento y despierto, atento y somnoliento, distraído y despierto, y distraído y somnoliento. Cuando un estudiante esté mostrando signos de distracción durante un tiempo prolongado, se le mostrará una alerta al profesor con la sugerencia de que adapte su metodología de enseñanza. Lo mismo ocurrirá con la somnolencia.

Para analizar la cara del usuario, se aplican técnicas de visión por ordenador que detectan rostros. Es importante señalar que los resultados de estos análisis serán estimaciones, pues la atención tiene una naturaleza compleja: engloba una combinación de características las cuales, muchas de ellas, no pueden ser capturadas solamente por una cámara.

La interfaz del usuario proporciona herramientas para administrar las sesiones de clase que se realicen, pudiendo así crearlas, o permitiendo eliminarlas. Esta interfaz se diseñará de manera que resulte usable y fácil de entender, para garantizar que cualquier tipo de profesor pueda utilizarla.

Con el fin de mantener la privacidad de los estudiantes, se implementarán medidas para anonimizar los datos, para así garantizar que la información recopilada se utilice únicamente con fines educativos y no se divulgue de manera inapropiada. Es por ello que al estudiantado no se le pedirá ningún tipo de información que le identifique más allá de un alias.

1.2. JUSTIFICACIÓN DEL PROYECTO

Este proyecto abarca campos de diferente índole, como son la ingeniería del software, el desarrollo de una aplicación web multimedia, el análisis y la minería de los datos, el almacenamiento de estos en una base de datos o la aplicación de la inteligencia artificial en la ejecución del servicio.

Estas características multidisciplinares son posibles gracias a las diferentes aptitudes, técnicas, conocimientos y habilidades que se aprenden en el Grado de Ingeniería Informática en Tecnologías de la Información de la UMH.

La evolución de las Tecnologías de la Información y la Comunicación (TIC) ha transformado la educación, permitiendo el aprendizaje remoto, pero también generando nuevos desafíos en la comprensión del proceso de aprendizaje. Esta solución software busca responder a las preguntas mencionadas previamente sobre qué es aquello que capta y mantiene la atención de los alumnos.

La Inteligencia Artificial permite que el sistema tome acciones para analizar la atención de manera similar a como lo haría una persona, facilitando esta tarea al profesor para que pueda centrarse en otras actividades. Además, una utilidad importante de este sistema es la generación de gráficas, que permiten analizar en qué momentos mantener la atención o despiertos a los alumnos ha supuesto una dificultad.

1.3. CONTEXTO E IMPORTANCIA

En este trabajo, se ha hecho uso de la Inteligencia Artificial para poder analizar el comportamiento de los usuarios. Tanto la inteligencia artificial como la ciencia de datos han evolucionado en los últimos años de manera exponencial. Estas tecnologías han sido debatidas y, debido a sus auges, han sido incluso reguladas en algunos aspectos por algunos gobiernos del mundo. Es tanta su extensión, que hoy en día existen muchos recursos para hacer uso de ellas de forma gratuita.

En esencia, la inteligencia artificial tiene como objetivo principal que uno o varios dispositivos tomen acciones y/o decisiones de manera similar a cómo lo haría una persona.

Este Trabajo Final de Grado utiliza la Inteligencia Artificial con el fin de analizar la atención de las personas haciendo uso de una cámara, facilitando la tarea a aquella persona que quiera medir la atención de modo que esta pueda focalizar sus esfuerzos en otras tareas. Además, extiende su funcionalidad permitiendo almacenar las diferentes sesiones en las cuales se ejecuta este software con la finalidad de su posterior estudio y análisis.

1.4. MOTIVACIONES PERSONALES

Este proyecto fue elegido de la lista de propuestas de los trabajos, y me resultó de interés debido a que abarcaba campos desafiantes y de actualidad como es la inteligencia artificial o la visión por ordenador. En mi opinión, estar en las últimas tendencias y trabajar con tecnologías actuales debería ser fundamental para cualquier ingeniero informático que desee expandir sus conocimientos y mejorar profesionalmente.

Además, para este trabajo se requiere de un amplio conocimiento en diferentes lenguajes y tecnologías de programación de aplicaciones web, por lo que todavía me resultaba más interesante desarrollar este proyecto, pues son conceptos que se abarcan poco en la carrera y cuyo grado de dificultad me resultaba atractivo. El trabajo contiene una diversidad de tecnologías amplia, lo cual despierta mi interés.

Otra motivación para realizar este trabajo, fuera del contexto de la Ingeniería Informática, fue el enfoque que tenía el proyecto. Considero que la educación debería ser el foco principal de cualquier sociedad, y ser capaz de contribuir en una herramienta que pueda ser usada para estudiar y mejorar aquello que nos hace prestar atención y, por tanto, aprender; así como también que pueda servir de inspiración para crear herramientas con la misma finalidad, me hizo sentir muy realizado y motivado.

1.5. LIMITACIONES DEL PROYECTO

Es importante considerar que este software no hace de sustituto a los métodos de conferencia online más populares, como podrían ser Google Meet o Zoom. Con este sistema se pueden crear salas online, y los alumnos se pueden unir a ellas, pero no se podrán ver las caras de los alumnos ni tampoco reconocer de ninguna manera quiénes son, a menos de que lo indiquen los alumnos explícitamente.

Pese a que es un sistema que graba y analiza en tiempo real, pretende ser anónimo, privado y seguro para los alumnos con el fin de obtener resultados de la atención no sesgados y lo más cercanos a la realidad.



Capítulo 2

Estado del arte



Biblioteca
UNIVER

2. ESTADO DEL ARTE

Para poder crear una aplicación que sirva para estudiar la atención, primero se debe definir y entender el concepto de “atención”, así como también se la debe descomponer en sus partes fundamentales. En este capítulo se desgranarán las partes principales de este proceso biológico y se profundizará en entender qué es la inteligencia artificial y cómo se puede emplear en este contexto.

2.1. INTELIGENCIA ARTIFICIAL

La inteligencia artificial es el campo de la informática que se especializa en realizar tareas que requieren inteligencia humana, como el aprendizaje, la percepción o el razonamiento. La Comisión Europea la define como sistemas de software y, posiblemente, hardware, diseñados por humanos que, ante un objetivo complejo, actúan percibiendo el entorno y razonando sobre el conocimiento [2].

Es un campo revolucionario que actualmente se encuentra en expansión y que está siendo utilizado en una amplia cantidad de situaciones y contextos, con diferente tipo de finalidades. Sin embargo, la inteligencia artificial viene de mucho tiempo atrás, concretamente en 1950, cuando Alan Turing publicó un artículo llamado “*Computing Machinery and Intelligence*” [3]. En este artículo, Alan Turing ya se preguntaba si las máquinas tendrían buen desempeño realizando tareas que se asemejen a la inteligencia humana. El matemático ya hablaba de pruebas que se le podrían realizar a una máquina con el fin de calificarla como una buena imitadora del comportamiento humano.

Debido a la naturaleza de su significado, las aplicaciones que tiene la inteligencia artificial están relacionadas en automatizar tareas relacionadas con la inteligencia humana: en el caso de este trabajo, la percepción, reconocimiento y análisis de la dirección de las cabezas, posturas y emociones.

2.1.1. APRENDIZAJE AUTOMÁTICO

El aprendizaje automático, comercialmente llamado *Machine Learning* (ML) es una aplicación de inteligencia artificial que permite que un sistema aprenda y mejore por su propia cuenta haciendo uso de redes neuronales y aprendizaje profundo, de modo que aprende gracias a grandes cantidades de datos [4].

Este aprendizaje va mejorando con el tiempo, pues conforme más medidas acumulen y, por tanto, más datos se introduzcan en el sistema, mejor funcionará, pues permitirá que el sistema identifique más patrones.

Este aprendizaje es un tipo de análisis predictivo, sin embargo, es mucho más fácil de implementar con actualizaciones en tiempo real a medida que se obtienen más datos. Es por ello por lo que es utilizado en este proyecto, pues uno de los requisitos del sistema es que sea en tiempo real.

2.1.2. REDES NEURONALES Y DEEP LEARNING

Las redes neuronales, o también conocidas como redes neuronales artificiales (ANN) o simuladas (SNN), son un subconjunto de aprendizaje automático. Tanto su nombre como su estructura se deben a las redes neuronales biológicas, y es que las ANN están formadas por capas de nodos que contienen una capa de entrada, una o varias capas ocultas, y una capa de salida. Estos nodos se conectan y tienen un peso y un umbral asociados. Los datos entre los nodos se envían siempre y cuando la salida de un nodo esté por encima del valor de umbral especificado, para así enviar datos a la siguiente capa de la red [5].

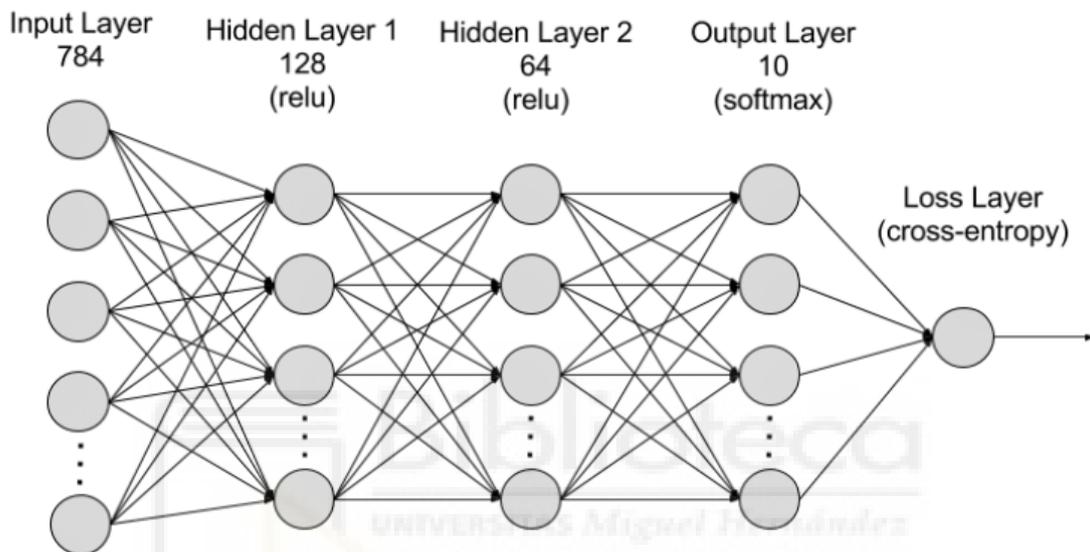


Figura 2.1—1 Ejemplo de una red neuronal [6]

Por otra parte, el aprendizaje profundo o *Deep Learning* (DL), como se suele conocer, es un subconjunto del aprendizaje automático que consiste, básicamente, en una red neuronal con tres o más capas [7]. Estas capas adicionales permiten una mayor precisión en los resultados finales.

2.1.3. LEGISLACIÓN SOBRE EL RECONOCIMIENTO FACIAL

La legislación referente a la inteligencia artificial y al reconocimiento facial de las emociones, o, en inglés, *Face Emotion Recognition* (FER), es, hoy en día, un tema de debate. Desde Europa, se pretende legislar estas tecnologías, con la intención de que una aplicación que las use deba cumplir los siguientes puntos [8]:

- **Necesidad y proporcionalidad:** El FER debe ser utilizado con cautela y cuando sea estrictamente necesario.
- **Precisión de la información:** Los datos deben ser considerados como imprecisos debido al contexto y a lo ambiguas que pueden ser las emociones, pues hay emociones que pueden expresarse a la vez o no llegar a ser expresadas del todo.
- **Justicia:** Se debe hacer uso de una buena base de datos que permita no discriminar por etnia ni condiciones físicas. Es común encontrar bases de datos que estén desbalanceadas y tengan más información sobre determinadas características físicas frente a otras.
- **Transparencia y control:** Se debe notificar y pedir consentimiento para procesar la información, incluido para compartirla con terceros.
- **No compartir categorías especiales de información personal:** es posible que se relacione el FER con otras categorías como la opinión política o condiciones de salud. Esta información no se puede usar por entidades no autorizadas, como podrían ser los empleados de una empresa.
- **No manipular a las personas:** no se debe usar FER para obtener información que pueda influir en opiniones o pueda ser usado para forzar a la gente a realizar acciones que no haría, así como tampoco pueden influenciar en el comportamiento.

Aunque en este proyecto no se analizan emociones, si se analiza la atención y la somnolencia, además de que se capturan las caras de las personas para estudiar estos estados mentales. Es importante tener la legislación vigente en cuenta para evitar problemas legales y morales tanto en la actualidad como cara al futuro.

2.2. INTRODUCCIÓN A LA ATENCIÓN

Según la Real Academia Española, la atención es la acción de *aplicar voluntariamente el entendimiento a un objeto espiritual o sensible* [9]. La motivación de este proyecto es la de estudiar y analizar cuándo se encuentra el individuo en este estado de entendimiento.

De manera más técnica, Estévez-González et al. [10] definen la atención como un estado neurocognitivo cerebral que precede a la percepción y a la acción y que permite al individuo seleccionar y mantener el foco sobre estímulos relevantes del entorno, ignorando aquellos irrelevantes. Al atender, se realiza un esfuerzo que afecta de manera significativa a la percepción, memoria y aprendizaje. Es por este motivo que es tan necesario saber qué produce que una persona esté atenta para poder enseñar de manera más eficiente.

Cada día, un individuo recibe estímulos de diversos tipos, manifestados de manera visual, auditiva, o de otras modalidades sensoriales. Estos estímulos exceden la capacidad del sistema nervioso para procesar todo en paralelo. Es por ese motivo por el que el organismo idea mecanismos como la atención para regularse y focalizarse.

2.2.1. TÉCNICAS DE VISIÓN POR COMPUTADORA PARA LA MEDICIÓN DE LA ATENCIÓN

En el ámbito educativo, se puede observar que la falta de atención de parte de los alumnos provoca que los estudiantes se encorven o apoyen sus cabezas [11]. La dirección de sus cabezas, sus emociones, sus posturas corporales y datos biológicos como la presión y la frecuencia cardíacas dan pistas del nivel de atención que pueden tener los alumnos, así como también nos proporcionan síntomas de desinterés o fatiga. En este caso, el proyecto se limita a la captura de imágenes que puedan estimar el nivel de atención de una persona y, por tanto, el resultado no será preciso en algunos casos.

2.2.1.1. SEGUIMIENTO DE LA MIRADA

El seguimiento de la mirada, o *gaze tracking*, es una técnica de visión por computadora que estima la dirección en la que una persona está mirando, infiriendo así su foco de atención [12]. Esta técnica se basa en el análisis de las imágenes de los ojos para determinar el punto de la mirada en una pantalla o en el entorno. Existen principalmente dos tipos de métodos de seguimiento de la mirada basados en visión por computadora: los basados en la apariencia, los basados en modelos y los basados en características.

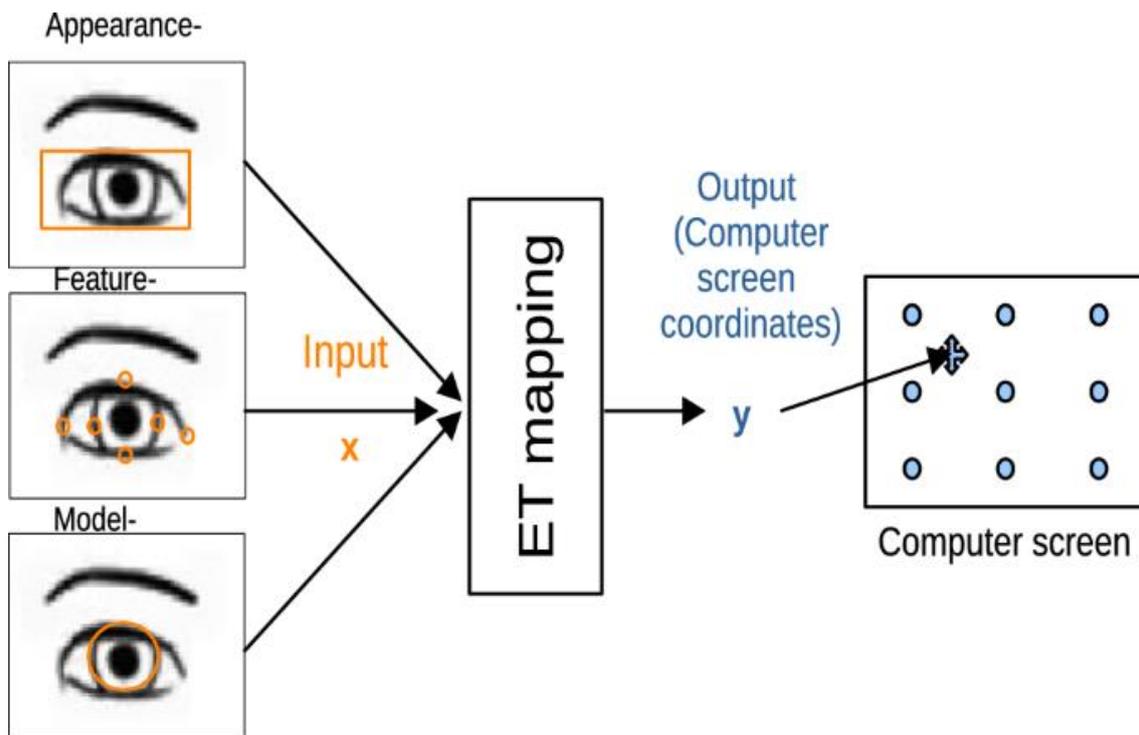


Figura 2.2—1 Métodos de seguimiento de la mirada

Los **métodos basados en la apariencia** mapean directamente las imágenes de los ojos o el rostro a la dirección de la mirada utilizando algoritmos de aprendizaje automático. Estos métodos, especialmente aquellos que emplean Redes Neuronales Convolucionales (CNNs), han demostrado ser muy efectivos para aprender las complejas relaciones entre la apariencia ocular y la dirección de la mirada.

Por otro lado, los **métodos basados en modelos** utilizan modelos 3D de la cabeza y/o los ojos para inferir la mirada a partir de parámetros geométricos. Estos métodos detectan la forma de los ojos (centro de la pupila, bordes del iris), la pose de los rasgos faciales y la orientación de la cabeza y los ojos para estimar la línea de visión 3D. Se han desarrollado sistemas de estimación de la mirada sin calibración, lo cual implica que el usuario no tiene que ajustar ningún parámetro. Esto facilita su implementación en diversos escenarios del mundo real.

Los **métodos basados en características** extraen rasgos específicos del ojo y del rostro, como el centro de la pupila, los bordes del iris o las esquinas de los párpados. A partir de estas características, calculan la dirección de la mirada mediante relaciones geométricas o modelos ligeros de aprendizaje automático. Estos métodos son más eficientes computacionalmente y pueden ofrecer una buena robustez ante condiciones variables, aunque su precisión depende en gran medida de la calidad en la detección de las características. A diferencia de los métodos basados en modelos, no usan un modelo geométrico en 3D, usan características en 2D. Son más simples y eficientes y, en casos concretos y complejos, menos precisos.

Raca et al. [13] determinan que se puede estimar el foco visual de la atención basándose, principalmente, en la postura de la cabeza, pues contribuye al 68.9% de la dirección de la mirada y constituye el 88.7% de precisión al determinar el foco de la atención. Estos autores señalan, también, que el recorrido que realiza la cabeza influye en gran medida en el nivel de atención.

La postura corporal también constituye gran parte de la atención que se presta. Es posible identificar cinco tipos de posturas principales en una clase: “atento”, “escribiendo”, “cabeza apoyada en la mano”, “inclinado hacia atrás”, y “sin mirar a la pantalla” [14]. En todos estos casos intervienen las manos, brazos, tronco... Sin embargo, debido a que el contexto del caso de uso es amplio y la cámara del usuario puede encontrarse en todo tipo de inclinaciones y posiciones (la cámara puede estar a un lado o de frente, arriba o abajo, estar grabando únicamente el rostro...) se ha considerado prescindible considerar la postura corporal, y únicamente se tendrá en cuenta si se está mirando la pantalla o no. Esto se hará considerando si se detecta un rostro o no.

En cuanto a las emociones, Paul Ekman definió que existen seis tipos de emociones básicas, las cuales son: alegría, asco, ira, miedo, sorpresa y tristeza [15]. Se puede asumir que, si un estudiante expresa uno o más tipos de emociones durante un umbral de tiempo establecido de manera frecuente y recurrente, no está prestando atención a las clases. Un estudio de la Universidad de California, Irvine y la Universidad de Humboldt concluyeron que el tiempo medio que le toma a una persona distraída volver al trabajo es de veintitrés minutos y quince segundos [16]. En este caso de uso, no se ha optado por el análisis de las emociones, pero sería importante considerarlo para ponderar la atención con ellas en futuras ampliaciones del proyecto.

Por las limitaciones computacionales que puedan tener los usuarios que utilicen la aplicación, las variabilidades en el contexto, las diferentes posiciones en las que se puede encontrar la cámara, y la flexibilidad que se persigue conseguir en el proyecto, se ha estudiado, probado y reflexionado en la metodología a aplicar y se ha podido llegar a la conclusión de que lo más óptimo, eficiente y seguro es utilizar un método basado en características donde se capturen las características de la cabeza para detectar la dirección de la mirada.

2.2.2. CÓMO MEDIREMOS LA ATENCIÓN EN ESTE TFG

Para poder detectar las imágenes que nos servirán para medir el grado de atención, se hace uso de una cámara. Esta cámara puede ser de cualquier tipo y será utilizada por los dispositivos de los alumnos.

Para detectar la atención en base a la dirección de la cabeza, se asume que el ángulo de la cabeza es donde está el foco de la atención. Si los estudiantes están mirando continuamente en una misma dirección, se entiende que dicho foco está concentrado en ese punto. Este foco no tiene por qué estar al frente: el usuario puede tener la pantalla o la cámara a un lado.

Es importante recordar que el contexto es muy importante en este tipo de situaciones. Se puede lograr aproximarnos a él de diferentes maneras como, por ejemplo, teniendo en cuenta el registro de datos pasados. Se puede dar el caso de que el estudiante se encuentre tomando apuntes o que gire la cabeza para mirar la proyección de las clases, la pizarra o para echar rápidamente un vistazo a otro lado. Si el foco cambia y se sostiene en un tiempo medianamente prolongado, se considera que se está centrado en la tarea. Sin embargo, si se está mirando hacia abajo con el fin de tomar apuntes o se echan vistazos rápidos, se tomarán como pequeñas distracciones que indicarán que la persona no está atenta.

2.3. INTRODUCCIÓN AL ESTADO DE SOMNOLENCIA

El sueño representa un estado fisiológico esencial, caracterizado por alteraciones cíclicas en la conciencia, la actividad muscular y la función cerebral, crucial para la restauración física y cognitiva [17].

La somnolencia, por su parte, se define como la tendencia de la persona a quedarse dormido. La medición de la somnolencia es compleja, y es por ello por lo que los medios que existen para medirla muestran poca concordancia entre ellos y muchos tienen un alcance limitado. Debido al contexto del proyecto, solo es posible ceñirse a un método que estudie el comportamiento para medir este estado. Para ello, se observa el pestañeo y el grado en la medida en la que se cierran los ojos para determinar si alguien se está durmiendo o no.

Analizar la somnolencia es importante en un contexto de clase virtual, pues nos da información de cómo está resultando la clase a los estudiantes o el estado en el que se encuentran, pues puede darse el caso de que la persona esté manteniendo su foco de atención en la clase, pero no esté prestando atención porque se está quedando dormida. Con información relevante al sueño de los estudiantes, el profesor puede considerar tomar medidas que capten también la atención de los estudiantes.

2.3.1. DETECCIÓN DE CARACTERÍSTICAS A PARTIR DEL PESTAÑO

El pestaño es la acción de abrir y cerrar los párpados y, aunque lo todas las personas lo hacen aproximadamente 15.000 veces al día y es un acto reflejo aparentemente simple, es fundamental para la protección y el mantenimiento de la salud ocular. [18]. El área de investigación que analiza la frecuencia y las características del pestaño, y los estados de sueño-vigilia, sugiere que los patrones de pestaño podrían reflejar el nivel de alerta, la presencia de fatiga y la calidad del sueño.

Existen tres tipos principales de pestaño: espontáneo, reflejo y voluntario [19]:

- El pestaño **espontáneo** ocurre de forma involuntaria, sin una causa externa aparente, y está relacionado con el mantenimiento de la humedad ocular y con funciones cognitivas.
- El pestaño **reflejo** se desencadena en respuesta a estímulos externos, como el contacto con la córnea o la aparición repentina de objetos cerca de los ojos, sirviendo principalmente para la protección ocular.
- El pestaño **voluntario** es aquel que se realiza de forma consciente y deliberada.

La existencia de estos diferentes tipos de pestaño indica que su frecuencia y características pueden ser debidas a diversos factores tanto internos como externos, lo cual es relevante para comprender su relación con el sueño y los estados de alerta.

2.3.1.1. PESTAÑO Y EL DESCANSO MENTAL

Más allá de estas funciones fisiológicas directas relacionadas con la salud ocular, el pestaño también desempeña roles importantes en las funciones cognitivas [20]. Se ha estudiado que el pestaño podría facilitar el "desenganche" de la atención. Tras el inicio de un pestaño, se observa una disminución de la actividad cortical en la red dorsal, que se asocia con la atención dirigida al exterior, y un aumento en la red de modo predeterminado, relacionada con el procesamiento interno y la introspección [21]. Esta alternancia en la actividad cerebral sugiere que el pestaño podría proporcionar pequeños momentos de descanso mental, permitiendo al cerebro procesar información de manera más eficiente y facilitar la reorganización atencional.

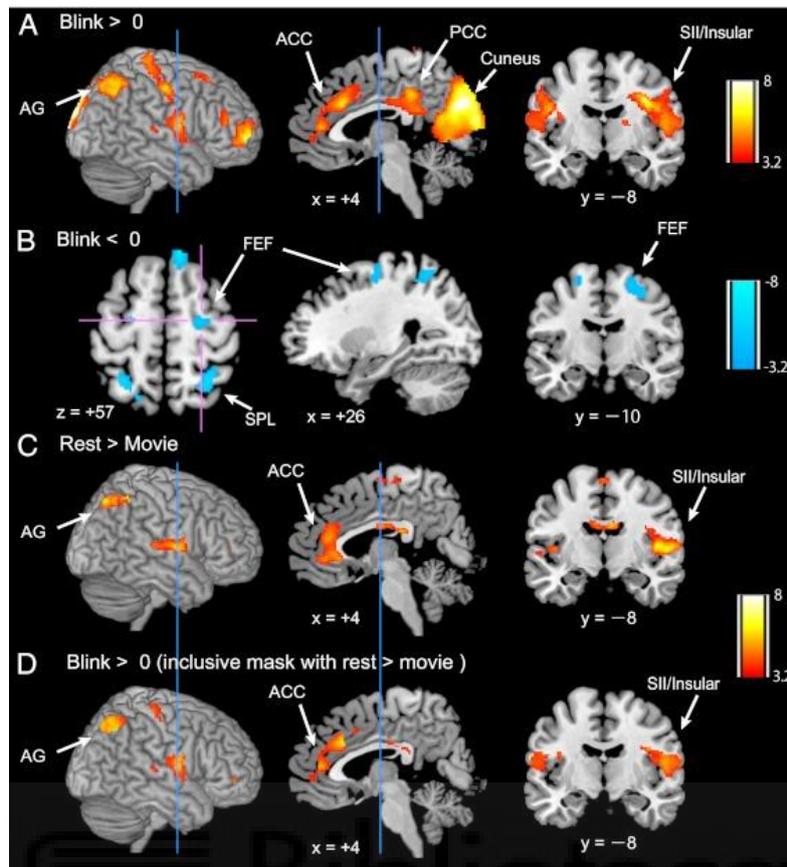


Figura 2.3—1 Relación de la activación cerebral con el parpadeo

En la Figura 2.3—1 Relación de la activación cerebral con el parpadeo se puede apreciar cómo diferentes regiones del cerebro se activan o desactivan en relación con los pestañeos espontáneos. Las dos primeras secciones, la A y la B, determinan, respectivamente, la activación y la desactivación de áreas del cerebro en momentos inmediatamente posteriores a un pestañeo.

En la primera sección, el panel A, se puede observar que, después de parpadear, se produce la activación de regiones que forman parte de lo que se denomina como “la red de modo predeterminado”, o DMN de sus siglas en inglés. Esta área suele activarse en estados de reposo o cuando el cerebro no está centrado en estímulos externos, como soñar despierto o pensar en uno mismo, por lo que sugiere que el pestañeo podría estar ligado a transiciones breves hacia procesos de pensamiento interno o reorganización cognitiva.

Por otro lado, en el panel B se muestran las áreas que disminuyen su actividad tras un pestañeo, como los campos oculares frontales (FEF) y el lóbulo parietal superior (SPL), regiones implicadas en el control de la atención visual dirigida hacia el entorno. Esta desactivación sugiere un "desenganche atencional" temporal, lo cual refuerza más todavía la idea vista en el primer panel de que el pestañeo sirve como una pausa cognitiva momentánea.

El panel C muestra la actividad cerebral en un estado de reposo mientras se ve una película. Las regiones que muestran mayor activación en reposo coinciden, en gran parte, con las que se activan tras un pestañeo, lo que refuerza la hipótesis de que parpadear puede actuar como una microtransición hacia un estado de reposo cognitivo.

Finalmente, el panel D combina las condiciones de activación por pestañeo (propias del panel A) y activación durante el reposo (del panel C) y muestra qué activaciones durante los pestañeos se solapan con las del estado de reposo. Lo que sugiere este panel es que algunas funciones visuales se suspenden brevemente, pero se mantiene una actividad introspectiva que podría facilitar la integración de nueva información o la reorganización atencional.

Por todo esto, se puede concluir que, si una persona parpadea mucho, esté perdiendo atención en el entorno y se encuentre más en sus propios pensamientos o esté cerca de dormirse pues, como se ha podido ver previamente, el sueño se caracteriza por esa ausencia de vigilia y pérdida de conciencia del entorno.

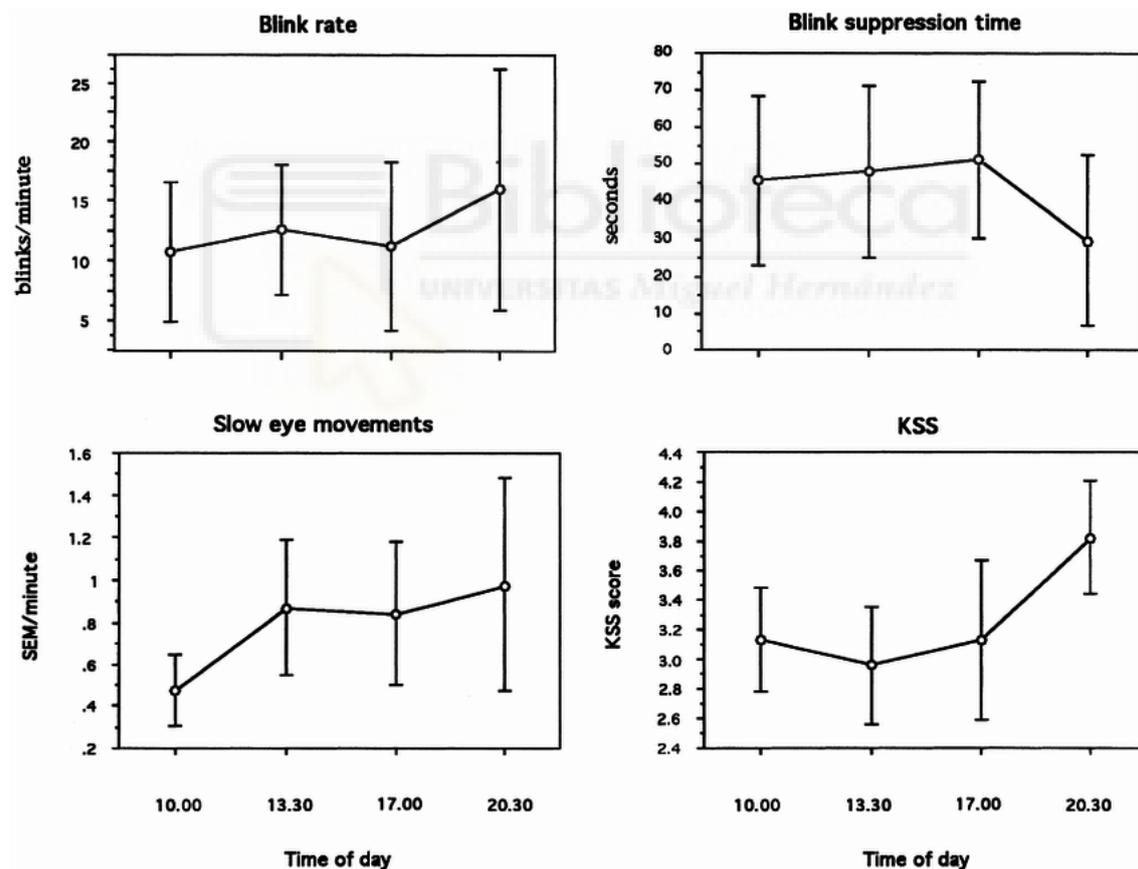


Figura 2.3—2 Variación de indicadores relacionados con la atención y somnolencia a lo largo del día

En la **¡Error! No se encuentra el origen de la referencia.** se representan cuatro métricas relevantes para el estudio de la atención y la somnolencia en diferentes momentos del día (10:00h, 13:30h, 17:00h y 20:30h): la tasa de parpadeo, el tiempo de

supresión de parpadeo, la frecuencia de movimientos oculares lentos y el nivel de somnolencia medido mediante la Escala de Somnolencia de Karolinska (KSS) [22].

- **Tasa de parpadeo** (*Blink rate*): Se observa un incremento progresivo en la cantidad de parpadeos por minuto conforme avanza el día, alcanzando su punto más alto a las 20:30h. Dado que una mayor frecuencia de parpadeo puede ser indicativa de fatiga y distracción, esta tendencia refuerza la hipótesis de que el cansancio afecta negativamente a la atención en las últimas horas del día.
- **Tiempo de supresión de parpadeo** (*Blink suppression time*): Este parámetro mide cuánto tiempo una persona puede mantener los ojos abiertos sin parpadear. El tiempo de supresión es más elevado entre las 13:30h y las 17:00h, disminuyendo de forma significativa a las 20:30h. Una menor capacidad para suprimir el parpadeo al final del día es otro signo de aumento de somnolencia.
- **Movimientos oculares lentos** (*Slow eye movements*): La frecuencia de movimientos oculares lentos, asociados a estados de fatiga y transición hacia el sueño, aumenta de forma gradual durante el día, alcanzando su pico máximo a las 20:30h. A pesar de que no se tiene en cuenta durante el análisis de la somnolencia y la atención de este proyecto, es interesante tenerlo presente en futuras ampliaciones.
- **Escala de Somnolencia de Karolinska** (*KSS*): El nivel subjetivo de somnolencia también sigue un patrón ascendente, siendo más bajo a media mañana (10:00h y 13:30h) y aumentando significativamente al final de la jornada. Podría ser interesante tener en cuenta como parámetro la hora del día a la cual se está realizando la clase para notificarlo al profesor o para analizar mejor la somnolencia en futuras ampliaciones del proyecto.

Estos resultados indican que tanto los parámetros fisiológicos (parpadeo, movimientos oculares) como las sensaciones subjetivas de somnolencia se ven afectados de manera consistente a lo largo del día. Este patrón es coherente con los ritmos circadianos humanos, que tienden a reducir el nivel de alerta en las horas vespertinas y nocturnas.

2.3.1.2. DETECCIÓN Y CÁLCULO DE UN PARPADEO

La detección de parpadeos se realiza a partir del seguimiento de los puntos correspondientes al párpado para ambos ojos. A partir de estos puntos se estimó el radio de aspecto para cada ojo (*EAR* – por sus siglas en inglés *Eye Aspect Ratio*), el cual refleja el estado de abierto o cerrado del ojo en base a la relación entre la distancia del párpado superior e inferior con respecto al largo del ojo, cumpliendo con la siguiente fórmula [23]:

$$EAR = \frac{|p2 - p6| + |p3 - p5|}{2|p1 - p4|}$$

Los puntos $p2$ y $p3$ son puntos del párpado superior, $p6$ y $p5$ puntos en el párpado inferior, y $p1$ y $p4$ puntos en los extremos del ojo, como se muestra en la Figura 2.3—3.

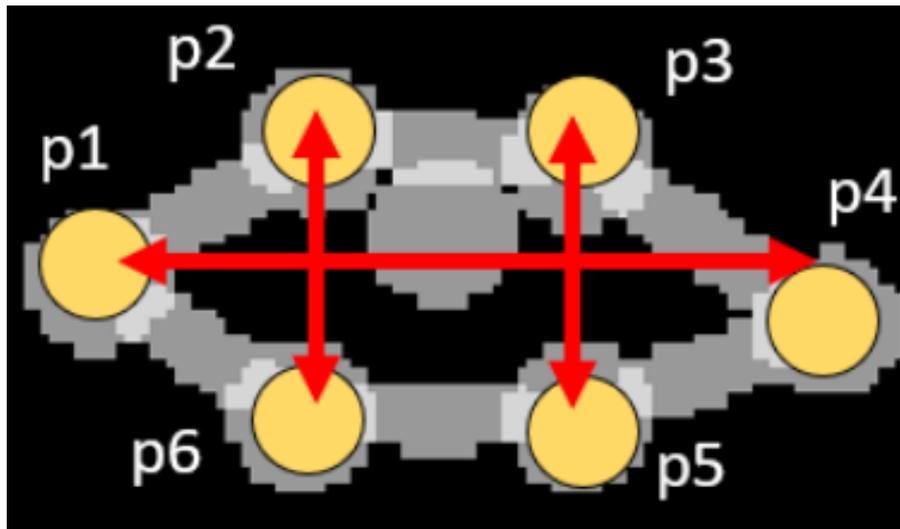


Figura 2.3—3 Puntos utilizados para estimar el EAR

Así, a menos distancia entre los párpados (es decir, cuanto más cerrados estén), el valor del EAR será menor y viceversa. Se puede hacer la media entre los valores de cada ojo para estimar cómo de abiertos tiene los ojos. Gracias a estos valores, es posible calcular la velocidad del cierre del ojo (*eye closure velocity*) para detectar los parpadeos [23].

La velocidad de cierre palpebral varía según si se trata de un parpadeo reflejo (30 ms) o un parpadeo espontáneo (75 ms), en el cual el retorno a la posición abierta es más lento, con un tiempo de 100-200 ms hasta la posición natural (Bahill et al., 1975) [24]. Si el retorno a la posición abierta es más largo que el mencionado previamente, se puede considerar que los ojos cerrados durante 1-2 segundos se asocian con un estado de "Somnolencia", durante 2-3 segundos con un estado de "Muy somnoliento" y durante más de 4 segundos "extremadamente somnoliento". En este proyecto, el grado de somnolencia ha sido simplificado a solo "somnoliento" y "despierto" a modo de agilizar el desarrollo y aligerar el coste computacional del sistema.

Es importante tener en cuenta también que la frecuencia media de parpadeo es de entre 15 y 20 pestañeos por minuto [25]. Valores por encima de esto suponen, también, indicios de somnolencia.



Capítulo 3

Hipótesis del trabajo

3. HIPÓTESIS DEL TRABAJO

En rasgos generales, el proyecto consiste en el desarrollo de una aplicación web que monitoriza, en tiempo real y de forma anónima, la atención y la somnolencia de los estudiantes en clases virtuales es posible mediante el uso de técnicas de visión por ordenador, inteligencia artificial y procesamiento de datos.

Se espera que, de esta manera, la aplicación proporcione información útil al profesorado para adaptar su metodología educativa, mejorando así la eficiencia en el proceso de enseñanza a los alumnos y, por tanto, también, su propio aprendizaje.

Es un servicio web que hará uso de cámaras habituales para funcionar. Se asume que cada cámara se sitúa justo enfrente de la cara del estudiante, encima de un monitor del ordenador de un aula; sin embargo, el sistema está adaptado para funcionar también en diferentes ángulos y todo tipo de dispositivos con cámara. Eso sí, es importante señalar que es un sistema que no funciona en remoto ya que el objetivo no es el de una herramienta que sirva de clase virtual sino una herramienta que asiste las clases impartidas. Esta cámara capturará imágenes del estudiante las cuales serán procesadas por el sistema para, posteriormente, almacenar su información más fundamental. Esta información será después interpretada y mostrada de manera gráfica en la interfaz del usuario.

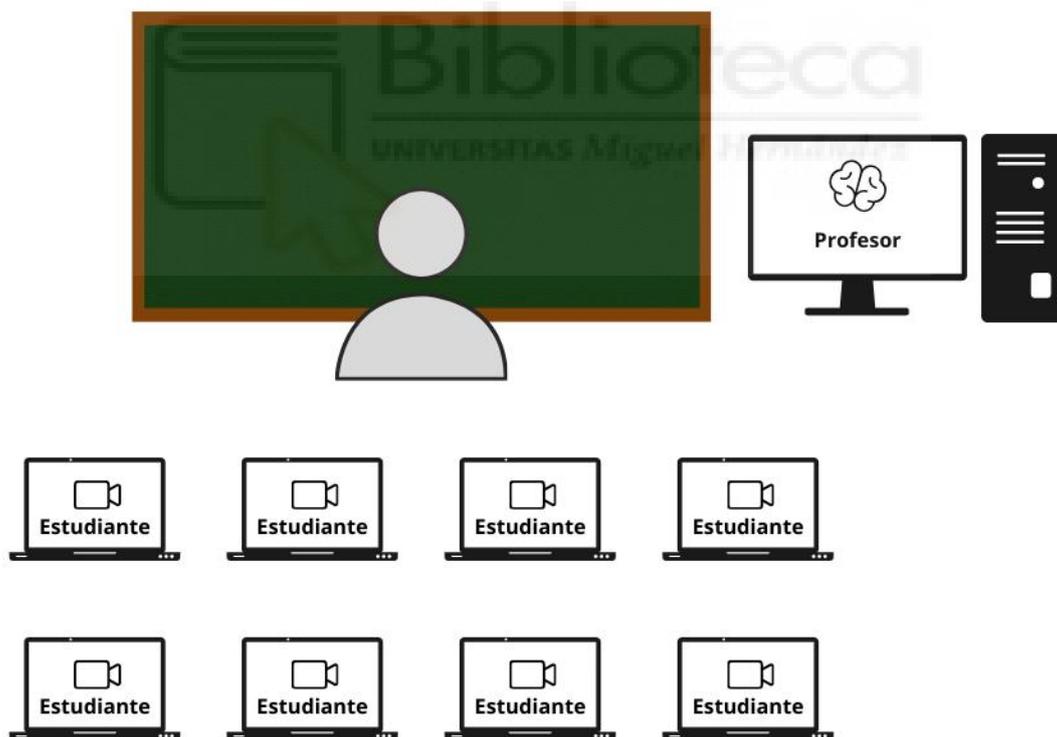


Figura 3—1 Contexto del uso del proyecto

La imagen representada en la Figura 3—1 Contexto del uso del proyecto muestra el ideal de cómo podría ser el funcionamiento de una clase habitual. Los estudiantes se sentarán en sus pupitres con unos dispositivos con cámara que ejecuten el software, y el profesor ejecutaría también el sistema para poder iniciar las grabaciones de la atención de las clases y poder ver las notificaciones.

Se busca el anonimato de los datos y se apunta a un uso ético de la aplicación, seguro y con fines investigadores con datos no sesgados: es por ello por lo que el sistema, a pesar de grabar a los usuarios, no almacena de manera permanente información personal sensible, como puede ser la cara de los usuarios o datos que le puedan identificar de alguna manera.

Como se ha mencionado previamente, un método basado en características resulta lo más útil, ágil, eficiente y sencillo para detectar la atención y la somnolencia en un contexto con tantas variables tan diferentes. Además, es posible aprovechar las características faciales obtenidas por medio de este tipo de método para obtener también los puntos relacionados con el radio de aspecto de los ojos y calcular, también, la somnolencia de los usuarios dentro del software. El objetivo general del software es detectar la posición de la cabeza y aproximarse a entenderla en su contexto para determinar si el usuario está prestando atención o no, mientras detecta, a su vez, si el usuario presenta signos de somnolencia determinados por la apertura de sus ojos y su frecuencia de parpadeo.

En este apartado se profundiza sobre el proceso de evolución del proyecto desde la idea inicial hasta el producto final, haciendo hincapié en el motivo por el que se ha escogido una opción frente a otras.

3.1. DISPOSITIVOS Y ESTRUCTURA

El sistema del proyecto depende de unos requisitos físicos tanto de parte del servidor como del cliente.

El servidor es aquél que realiza el papel de distribuidor del sistema y entidad que almacena la información y ejecuta la lógica de las sesiones del sistema.

El cliente, por su parte, realiza el papel de anfitrión o el de participante de las salas, es el usuario final del sistema y ejecuta la lógica de análisis de la atención, el envío de los datos al servidor y el consumo y uso de la información almacenada en la base de datos del sistema.

Para el desarrollo de las pruebas, se ha utilizado un mismo ordenador como cliente y como servidor, y ha podido funcionar sin ningún tipo de problema con números reducidos de usuarios.

3.1.1. CARACTERÍSTICAS DEL SERVIDOR

El computador utilizado para el desarrollo y despliegue del programa, tanto en su versión final como en sus versiones de pruebas, cumple con las siguientes especificaciones:

Tabla 3.1—1 Especificaciones del ordenador

| | |
|--|------------------------|
| Sistema operativo | Windows 11 64 bits |
| Unidad de procesamiento (CPU) | Intel Core i5-8250U |
| Tarjeta de procesamiento gráfico (GPU) | Intel UHD Graphics 620 |
| Memoria principal | 8GB |
| Memoria secundaria | 500GB |

El servicio debería poder funcionar en ordenadores de características similares o superiores a las expuestas. Con estos requisitos el sistema ya es capaz de soportar varios profesores con sus respectivas clases ejecutándose y varios alumnos diferentes accediendo y siendo analizados a la vez.

3.1.2. CARACTERÍSTICAS DEL CLIENTE

El cliente podrá acceder desde cualquier dispositivo que cuente con un navegador web y una cámara. El sistema está diseñado y optimizado para que funcione en una variedad amplia de dispositivos, y es por ello por lo que los recursos no son altos. Funciona en cualquier teléfono móvil, tableta u ordenador de hoy en día y con cualquier tipo de navegador.

Para la captura de las imágenes, se hace uso de una cámara cualquiera que esté disponible en el dispositivo el cuál vaya a actuar como cliente del servicio. No es necesario que las características de la cámara sean muy buenas: con que pueda capturar imágenes con una resolución de 480 píxeles de alto y 640 píxeles de ancho debería ser suficiente. Lo ideal es que capture en 720x1280 píxeles. Las imágenes más grandes que el software permite capturar son de 1080x1920 píxeles.

3.2. LENGUAJES DE PROGRAMACIÓN Y LIBRERÍAS

Los lenguajes de programación y tecnologías utilizados en el proyecto han sido elegidos teniendo en cuenta que el producto final se trata de una aplicación web que aplica técnicas de inteligencia artificial. En este apartado se especifican las tecnologías software empleadas en el proyecto, así como una breve explicación de qué ofrecen y por qué se han usado.

3.2.1. SERVICIO WEB

Al necesitar un servicio web, es necesario hacer uso de lenguajes, librerías y sistemas orientados a este campo de la ingeniería informática. Estas tecnologías tienen la premisa de presentar de manera visual, estética y sencilla las interpretaciones del servicio. Este apartado detalla las tecnologías fundamentales y más relevantes para la creación de la interfaz web de la aplicación, cuyo objetivo es presentar de manera visual y accesible los resultados del análisis de atención.

3.2.1.1. HTML

HyperText Markup Language, o “Lenguaje de Marcado de Hipertextos” es el lenguaje de marcado estándar que define el significado y la estructura del contenido de la web. Este lenguaje permite marcar los diferentes tipos de multimedia para mostrarlos en un navegador web. También permite especificar, dentro de cada tipo de multimedia, de qué subtipo se está hablando [26].

Debido a que es un elemento básico para estructurar el contenido de la web, en este proyecto se ha empleado esta tecnología para definir la estructura y el significado del contenido que se mostrará al profesor, como los paneles de visualización de datos, las listas de sesiones y los diferentes formularios, todo con el fin de poder mostrar el contenido.

3.2.1.2. CSS

Cascading Style Sheets, o “Hojas de Estilo en Cascada”, es un lenguaje de estilo usado para describir cómo se presentan los documentos basados en XML, como el lenguaje HTML. Es utilizado para diseñar y dar estilo a los sitios web, entre otros tipos de ámbitos, y permite alterar de manera estética cómo se presenta la información [27].

Su uso en este proyecto se centra en cumplir con el requisito no funcional de “diseño estético, minimalista y sencillo”, que se encontrará más adelante en el capítulo de “Requisitos no funcionales”. CSS permite controlar la apariencia de los elementos web, como colores, fuentes y su disposición.

3.2.1.3. TAILWIND

Es un *framework* (en español, marco de trabajo) de CSS que facilita la aplicación de estilos mediante clases predefinidas directamente en los elementos HTML. Sirve para, haciendo uso de clases en nuestros elementos HTML, darle estilo a nuestro sitio web. A pesar de no ser muy partidario de utilizar únicamente Tailwind para darle estilo al sitio, sí es útil para agilizar la tarea en ciertos elementos más pequeños del sitio web [28]. Como todo *framework*, considero que se debería usar cuando resulte útil para el desarrollo de la aplicación, y no como la tónica general del proceso de creación, y su justificación radica en su utilidad para un desarrollo más rápido en ciertos aspectos de la interfaz.

3.2.1.4. JAVASCRIPT

JavaScript es un lenguaje de programación, principalmente para *scripting* (secuencias de comandos) para páginas web, aunque también es usado en entornos fuera de él. Es multiparadigma, de un solo hilo, dinámico y con soporte para programación orientada a objetos, imperativa y declarativa [29]. Resulta esencial para la interactividad en las páginas web.

Debido a que se usará la construcción de interfaces basada en componentes de React, y esta librería está basada en JavaScript, será imprescindible utilizar este lenguaje para la creación de la página a partir de piezas individuales.

3.2.1.5. TYPESCRIPT

TypeScript es un lenguaje de programación que actúa como superconjunto de JavaScript, extendiendo su sintaxis. Es por ello por lo que cualquier código JavaScript funcionará en TypeScript. Resulta de gran utilidad porque otorga a JavaScript la capacidad de darle tipos a las variables [30].

Se usará con la finalidad de darle robustez a la aplicación en tiempo de ejecución y aprovechar las ventajas que proporciona otorgar de tipado a las variables.

3.2.1.6. REACT

React es una biblioteca de JavaScript que permite crear interfaces de usuario, tanto en la web como de forma nativa. React se basa en crear componentes y combinarlos para formar páginas y aplicaciones. Es una biblioteca que permite agrupar componentes, pero no permite el desarrollo *full-stack* de la web. Para ello se necesitará un *framework* como el de Next.js, tratado en el siguiente apartado.

En React, los componentes se crean haciendo uso de HTML, CSS y JavaScript. Estos componentes pueden tener una serie de propiedades que permiten ser definidas en las distintas instancias del componente dentro de la aplicación. Esta biblioteca nos permite reutilizar una gran cantidad de código, así como también escribirlo de modo mucho más modular y siguiendo un paradigma orientado a objetos. Esto es especialmente útil en interfaces donde el contenido sigue un determinado patrón y es muy dinámico [31].

En este proyecto, estos componentes se hacen más claros a la hora de mostrar patrones como la disposición de la cabecera o la presencia o no de una barra lateral, los botones, las entradas de datos, así como también otros elementos. Estos serán componentes que podrán ser reutilizados.

Debido a que la idea es crear un servicio con componentes para su eficiente implementación, y se hará un desarrollo *full-stack*, se precisará de React para usar Next.js.

3.2.1.7. NEXT.JS

NextJS es un *framework* de React que sirve para crear aplicaciones web. Este *framework* proporciona todas las herramientas necesarias para crear la interfaz de usuario y comunicarte con el servidor y la base de datos.

NextJS permite actualizar fácilmente el código HTML del servicio web, así como también soporta estilos CSS. Además, facilita el manejo de servicios de terceros, permite generar autenticaciones de los usuarios, da acceso a obtener datos del servidor de manera rápida y eficiente, permite ejecutar código de servidor de manera sencilla, sirve para actualizar la interfaz de usuario de manera rápida y sin complicaciones, y facilita la creación de rutas avanzadas usando el sistema de archivos [32].

Se usará NextJS para poder comunicar el servicio web con la base de datos y acceder a toda la parte de servidor, así como también poder hacer uso de los componentes de React.

3.2.1.8. NODE.JS

Node es un entorno de ejecución que permite que Javascript pueda funcionar en un modo multiplataforma. Este entorno permite crear servidores, aplicaciones web o herramientas de líneas de comando y scripts. Es una tecnología necesaria para el desarrollo de aplicaciones en NextJS. Se utiliza su gestor de paquetes NPM (*Node Package Manager*) para instalar y actualizar los *frameworks*, librerías y paquetes necesarios [33].

Se usará NodeJS para poder crear el proyecto en Next y poder tenerlo actualizado, así como también tener actualizados todos los paquetes. También servirá para ejecutar el servidor.

3.2.1.9. MYSQL

MySQL es el sistema de gestión de bases de datos relacional más popular del mundo [34]. Gracias a este sistema, es posible almacenar y hacer uso de la información que se recopile en una aplicación.

Es fundamental añadir un sistema de gestión de bases de datos para poder añadir interactividad y persistencia de los datos en el sistema.

3.2.1.10. SQL

SQL, o *Structured Query Language*, es un lenguaje de consulta estructurado con el propósito de manipular y almacenar datos de una base de datos relacional [35].

Este lenguaje de programación resulta imprescindible si se quiere utilizar el sistema de gestión de base de datos MySQL.

3.2.1.11. SOCKET.IO

Socket.IO es una librería que permite una comunicación de baja latencia, bidireccional y basada en eventos entre cliente y servidor. Esta conexión es flexible: en lugar de depender de un único método, puede establecerse en diferentes niveles de transporte para garantizar un uso de la conexión seguro, veloz, fiable y adaptable a diferentes entornos [36].

Un socket es un espacio de conexión virtual entre dos dispositivos. A través de este punto se pueden enviar y recibir datos. Por ello, es útil una librería que facilite la comunicación entre cliente y servidor para poder crear la aplicación y que funcione en tiempo real.

3.2.1.12. PRISMA

Prisma es una librería que simplifica y facilita el acceso a la base de datos y la manipulación de esta, como las conexiones a la base de datos, el almacenamiento en caché, las inserciones, modificaciones y eliminaciones de datos, entre otros [37]. Es una librería necesaria si se quiere trabajar de manera veloz y sencilla usando NextJS para acceder a las bases de datos de MySQL.

3.2.2. TECNOLOGÍAS PARA LA INTELIGENCIA ARTIFICIAL

3.2.2.1. MEDIPIPE

MediaPipe es un conjunto de bibliotecas y herramientas que sirve para aplicar la inteligencia artificial rápidamente en las aplicaciones. Esta librería contiene diferentes soluciones para distintos casos de uso. La utilizada en este proyecto es la solución de “FaceMesh”, una solución que permite detectar la cara de una persona, dándonos 468 puntos de interés que se puede utilizar para cualquier conveniencia [38].

Se hace uso de MediaPipe ya que el proyecto captura la mirada, el pestañeo y demás información relacionada con la gesticulación facial para el análisis de la atención y la somnolencia.

3.2.3. JUSTIFICACIÓN DE LAS TECNOLOGÍAS

La elección de las tecnologías utilizadas en el desarrollo de este proyecto responde tanto a criterios técnicos como prácticos, teniendo en cuenta la escalabilidad, facilidad de mantenimiento, experiencia previa del desarrollador y la comunidad de soporte existente. Se busca construir una solución robusta, escalable, segura, multiplataforma y con soporte en tiempo real.

La aplicación se ha desarrollado por web para que la única tarea del usuario sea la de conectarse al servicio. NextJS ha sido seleccionado como *framework* principal por su capacidad de integrar frontend y backend en un mismo entorno, simplificando el desarrollo full-stack. Es un marco de trabajo muy utilizado hoy en día y, por tanto, con mayor documentación y recursos. Ofrece un tiempo de carga muy pequeño, con sitios web adaptables según el tamaño de la pantalla y que funcionan en cualquier dispositivo,

y garantiza la seguridad de los datos [29]. Se optó por NextJS frente a otros marcos de trabajo debido, principalmente, a los conocimientos previos del *framework* y a su correspondiente familiaridad con la tecnología. Su integración con React permite un desarrollo basado en componentes, ideal para interfaces dinámicas.

React, como biblioteca de interfaz de usuario, ha sido clave para presentarle los elementos al usuario de manera modular y reutilizable. Su modelo basado en componentes facilita el mantenimiento del código y su implementación de un DOM virtual optimiza la renderización de cambios en la interfaz.

TypeScript se ha utilizado para mejorar la robustez del código. Su sistema de tipos permite detectar errores en tiempo de desarrollo, lo cual facilita el mantenimiento y la escalabilidad del proyecto.

Tailwind CSS ha sido empleado para agilizar la maquetación visual con clases utilitarias predefinidas, permitiendo un diseño rápido, minimalista y adaptable, clave para cumplir con el requisito no funcional de estética y usabilidad.

Node.js y NPM han permitido configurar y gestionar el entorno de desarrollo, así como instalar y mantener todas las dependencias necesarias del proyecto.

MySQL, como sistema de gestión de bases de datos relacional, ha sido seleccionado por su fiabilidad, rendimiento y compatibilidad con Prisma y Next.js. Su lenguaje SQL proporciona una estructura clara y potente para la manipulación de datos.

Prisma ORM se ha utilizado para simplificar el acceso a la base de datos desde el backend. Permite generar consultas complejas de manera intuitiva y garantiza el tipado gracias a su integración con TypeScript.

Socket.IO ha sido la tecnología elegida para permitir comunicación bidireccional entre cliente y servidor, necesaria para enviar datos de atención en tiempo real. Su flexibilidad y facilidad de uso la hacen ideal para aplicaciones que requieren respuesta inmediata sin recargar la página.

MediaPipe ha sido esencial, ya que proporciona una solución eficaz y ligera para la detección de puntos faciales en tiempo real, lo que ha permitido implementar el análisis de atención y somnolencia sin necesidad de entrenar modelos desde cero. Su modelo de FaceMesh ofrece una precisión suficiente para los objetivos del proyecto, y su integración en el navegador facilita el procesamiento directamente en el cliente, respetando así la privacidad y reduciendo la carga del servidor.

Además de los motivos individuales, todas estas tecnologías comparten características que las hacen ideales para este proyecto:

- Son de código abierto, esto permite mayor accesibilidad que las que no lo son.
- Tienen amplia documentación y una comunidad activa, lo que ha facilitado en gran medida el aprendizaje de las tecnologías, la resolución de errores que se han ido presentando y podría garantizar un correcto mantenimiento en un futuro.
- Son modulares y escalables, permitiendo futuras ampliaciones del sistema sin necesidad de rehacer su arquitectura.
- Permiten que la aplicación funcione correctamente en distintos dispositivos.

Capítulo 4

Metodología y resultados



4. METODOLOGÍA Y RESULTADOS

A continuación, se detalla la metodología seguida para el desarrollo de la aplicación. El proyecto se abordó siguiendo un modelo de ciclo de vida estilo Kanban, el cual se describe en la siguiente sección y sirvió como guía para las diferentes etapas del desarrollo, desde la concepción inicial hasta la obtención de los resultados que se presentan en esta memoria.

4.1. CICLO DE VIDA

En este apartado, se explica qué es la metodología Kanban empleada en el ciclo de vida de este proyecto y qué ventajas tiene en un proyecto individual que permiten que sea un sistema rápido, sencillo y eficiente para desarrollar el producto.

4.1.1. METODOLOGÍA KANBAN

En el desarrollo de este proyecto se ha optado por utilizar la metodología Kanban como sistema de gestión del flujo de trabajo debido a que este método se basa en un mecanismo de control visual que permite representar y seguir el avance de las tareas a través de las distintas fases del proyecto. Para ello, se emplea un tablero dividido en columnas como "Por hacer", "En progreso" y "Hecho", sobre el que se colocan tarjetas que representan cada tarea específica. En el caso de este proyecto, se ha realizado en formato digital utilizando la herramienta online *Trello* [39].

Aunque Kanban se utiliza frecuentemente en equipos de trabajo, su implementación en proyectos individuales también resulta altamente beneficiosa. Entre las principales razones para aplicar este enfoque en un proyecto individual frente a otras metodologías, se destacan:

- **Visualización clara del estado del proyecto:** permite tener una vista rápida y ordenada de qué tareas están pendientes, en desarrollo o completadas, lo cual mejora la planificación personal.
- **Foco en el trabajo en curso:** Kanban promueve limitar el número de tareas en paralelo con el lema de “*Stop Starting, start finishing*” (Deja de empezar, empieza a terminar) ayudando a mantener la concentración y reducir la multitarea, lo que resulta en una mayor eficiencia.
- **Seguimiento y autogestión del tiempo:** facilita la organización del trabajo a lo largo del tiempo, permitiendo identificar cuellos de botella o retrasos.
- **Flexibilidad y adaptación al cambio:** permite organizar tareas o cambiar prioridades fácilmente.

Las tres reglas de Kanban es que debe:

1. Mostrar el proceso.

Consiste en la visualización de todo el proceso de desarrollo mediante el tablero anteriormente mencionado. El objetivo de ello es entender mejor el proceso de trabajo actual y conocer los problemas que puedan surgir con el fin de tomar decisiones.

2. Limitar el trabajo en curso.

Los límites del trabajo en curso consisten en acordar previamente la cantidad de tareas que se pueden realizar en cada proceso, con el fin de detectar dónde se está estancado y seguir la premisa anterior de terminar tareas y no empezar muchas.

3. Optimizar el flujo de trabajo

El objetivo es generar una producción estable, continua y previsible. Si se mide el tiempo que se tarda en realizar una tarea se puede medir también el rendimiento que se tiene por tarea para así estimar en cuánto tiempo se puede realizar un objetivo del proyecto si se divide en objetivos más pequeños y abordables.

Por estas razones, Kanban representa una herramienta de gran utilidad para gestionar de manera eficiente el desarrollo de este proyecto, pues al trabajar de forma individual se consigue aportar claridad, orden y control en todas las fases del proyecto. Ha ayudado mucho a identificar los cuellos de botella y a tener claros, en todo momento, los objetivos del proyecto [40].

4.2. ESPECIFICACIÓN

Para la especificación de requisitos y casos de uso del sistema se ha optado por una metodología basada en ingeniería de requisitos de prototipado evolutivos (*Evolutionary prototyping*). Esta metodología consiste en la elaboración de prototipos con modelos funcionales del sistema (o parte de él) en base a requisitos generales y a las evaluaciones de los usuarios. En este caso, el de un trabajo de fin de carrera, es el tutor del grado y el estudiante quienes evalúan los modelos [41]. Es un sistema bueno cuando se tiene una idea general de los requisitos, pero no se saben de manera precisa.

Esta elección se debe a que ha sido un proyecto desarrollado por una única persona y, además, que se ha ido aprendiendo acerca del caso de uso y de las tecnologías a medida que se ha ido desarrollando.

En este apartado, se describirán los requisitos y funcionalidades del software que se ha desarrollado, así como también los casos de uso que se precisarán y los actores que se verán involucrados en los mismos.

4.2.1. REQUISITOS DEL SOFTWARE

Los requisitos del software son las necesidades, restricciones o requerimientos que un software debe cumplir para que se considere como válido debido a que se alinea con los deseos o necesidades del cliente para el que está destinado. En otras palabras, los

requisitos definen qué debe hacer el software, cómo debe verse y las condiciones que deben cumplirse para que se considere exitoso. Se distinguen dos tipos de requisitos: funcionales y no funcionales [42].

En este apartado, se especifican cuáles son los requisitos que cumple el software del proyecto.

4.2.1.1. REQUISITOS FUNCIONALES

Los requisitos funcionales son aquellos que constituyen una parte fundamental para la ejecución del programa. Tienen una descripción clara de cómo se espera que el sistema responda a un comando en particular, las características que debe cumplir y lo que esperan los usuarios que ocurra.

Tabla 4.2—1 Requisito funcional: Captura de la dirección de la cabeza

| | |
|--------------------|--|
| RF-1 | Captura de la dirección de la cabeza |
| Descripción | La aplicación será capaz de capturar y almacenar el grado de inclinación de la cabeza y su posición en el espacio. |

Tabla 4.2—2 Requisito funcional: Detección de parpadeos y ojos cerrados

| | |
|--------------------|--|
| RF-2 | Detección de parpadeos y ojos cerrados |
| Descripción | La aplicación será capaz de determinar, gracias a la detección de la apertura del ojo, si se parpadea o no, así como también cuánto se cierra el ojo. Podrá contemplar la posibilidad de que los ojos estén cerrados, entreabiertos, abiertos y con qué frecuencia se abren y cierran. |

Tabla 4.2—3 Requisito funcional: Medición del tiempo de la atención

| | |
|--------------------|--|
| RF-3 | Medición del tiempo de la atención |
| Descripción | Haciendo uso de la posición de la cabeza, se determinará si el usuario mantiene el foco en un mismo punto y, por tanto, si presta atención, y durante cuánto tiempo la ha sostenido. |

Tabla 4.2—4 Requisito funcional: Medición del tiempo de la somnolencia

| | |
|--------------------|---|
| RF-4 | Medición del tiempo de somnolencia |
| Descripción | Haciendo uso de la información de la apertura del ojo, se determinará si el usuario se está durmiendo y durante cuánto tiempo lo está haciendo. |

Tabla 4.2—5 Requisito funcional: Grabación de las sesiones

| RF-5 | Grabación de las sesiones |
|--------------------|---|
| Descripción | Se podrán grabar las sesiones en las que participe el usuario y se almacenarán para poder ser visualizadas más tarde. Estas grabaciones no serán de vídeo con el fin de mantener el anonimato, únicamente se registrarán los cambios en la atención o vigilia de los estudiantes, el momento en el que han ocurrido estos cambios y el tiempo que ha durado la clase. |

Tabla 4.2—6 Requisito funcional: Sistema en tiempo real

| RF-6 | Sistema en tiempo real |
|--------------------|---|
| Descripción | El sistema debe capturar y mostrar los resultados en tiempo real, y debe mostrarlos, también en tiempo real, por medio de una interfaz gráfica. |

Tabla 4.2—7 Requisito funcional: Administración de las sesiones

| RF-7 | Administración de las sesiones |
|--------------------|---|
| Descripción | Debe existir un mecanismo para poder visualizar cada una de las sesiones pasadas, así como también su información. También debe haber la posibilidad de borrar la información referente a ella. |

Tabla 4.2—8 Requisito funcional: Administración del perfil

| RF-8 | Administración del perfil |
|--------------------|--|
| Descripción | Debe existir un mecanismo para poder visualizar la información relevante al perfil del profesor, así como borrar su cuenta o editar su alias o contraseña. |

Tabla 4.2—9 Requisito funcional: Resultados globales

| RF-9 | Resultados globales |
|--------------------|---|
| Descripción | Se podrán visualizar los resultados obtenidos por cada una de las sesiones que se hayan realizado. Cada informe mostrará su información de manera global. |

Tabla 4.2—10 Requisito funcional: Resultados individuales

| RF-10 | Resultados individuales |
|--------------------|--|
| Descripción | Dentro de cada sesión, será posible ver detalles relevantes a la atención por cada alumno en individual. |

Tabla 4.2—11 Requisito no funcional: Notificación al profesor de la ausencia de atención

| | |
|--------------------|---|
| RF-11 | Notificación al profesor de la ausencia de atención |
| Descripción | La ausencia de atención por parte de uno o varios alumnos se le será notificada en el momento al profesor. Esta notificación sonará y podrá ser silenciada para no causar molestia. |

Tabla 4.2—12 Requisito no funcional: Notificación al profesor de la somnolencia

| | |
|--------------------|---|
| RF-12 | Notificación al profesor de la somnolencia |
| Descripción | La presencia de sueño por parte de uno o varios alumnos se le será notificada en el momento al profesor. Esta notificación sonará y podrá ser silenciada para no causar molestia. |

4.2.1.2. REQUISITOS NO FUNCIONALES

Los requisitos no funcionales son aquellas restricciones o necesidades que resultan meramente estéticas o destinadas para otras funciones que no formen parte de la esencialidad del proyecto. Si los requisitos funcionales especifican lo que debe hacer un sistema, los requisitos no funcionales describen cómo lo hará el sistema.

Tabla 4.2—13 Requisito no funcional: Diseño estético, minimalista y sencillo

| | |
|--------------------|---|
| RNF-1 | Diseño estético, minimalista y sencillo |
| Descripción | La interfaz será agradable a la vista y mostrará el contenido mínimo e indispensable. Será sencilla de aprender, con iconografía simple y una interfaz que muestre su funcionamiento de manera intuitiva. |

Tabla 4.2—14 Anonimato de los alumnos

| | |
|--------------------|---|
| RNF-2 | Anonimato de los alumnos |
| Descripción | La identidad de los alumnos debe ser protegida y no debe ser posible identificarles de ninguna manera. La cámara ser únicamente para evaluar el estado de atención del alumno y los datos no deben poder relacionarse con el individuo. |

Tabla 4.2—15 Requisito no funcional: Código escalable

| | |
|--------------------|--|
| RNF-3 | Código escalable |
| Descripción | El código debe ser reutilizable y debe permitir poder ser ampliado con facilidad en un futuro. |

Tabla 4.2—16 Requisito no funcional: Código modular

| | |
|--------------------|---|
| RNF-4 | Código modular |
| Descripción | El código debe estar separado en módulos distintos que realicen funciones claras y bien definidas para su fácil ampliación, mejora, utilización y depuración. |

Tabla 4.2—17 Requisito no funcional: Librerías de código abierto

| | |
|--------------------|--|
| RNF-5 | Librerías de código abierto |
| Descripción | El código perteneciente a las librerías empleadas en esta aplicación web deben ser de código abierto para que resulte económicamente rentable. |

4.2.2. ACTORES

Se le llaman “actores” a los tipos de usuarios que interactúan con una aplicación. En este apartado, se expondrán qué actores existen en el proyecto y qué casos de uso o acciones pueden realizar dentro del mismo.

Tabla 4.2—18 Actor: Profesor

| | |
|--------------------|---|
| Actor | Profesor |
| Descripción | Es el anfitrión de la clase, y tiene la capacidad de uso activo de la totalidad la aplicación. Puede crear sesiones de grabación, administrar su cuenta, ver los resultados del análisis de sus sesiones, terminar las clases y también eliminar las clases de su perfil. |

Tabla 4.2—19 Actor: Estudiante

| | |
|--------------------|---|
| Actor | Estudiante |
| Descripción | Es el participante de la clase y, a pesar de no tener control sobre la aplicación, genera las entradas básicas al sistema; es decir, genera la información de la mirada y la información relevante a la dirección donde tiene puesto el foco de su atención. Puede unirse a las clases y envía datos al servidor. |

Tabla 4.2—20 Actor: Inteligencia Artificial (IA)

| | |
|--------------------|--|
| Actor | Inteligencia Artificial (IA) |
| Descripción | Se considera a la IA como un actor debido a su capacidad de identificación de patrones, similares a las que haría un humano. Actúa como intermediario y constituye una parte fundamental del sistema, pues es el encargado de procesar la información del estudiante para facilitar el conocimiento al profesor. |

Todos estos actores realizan una serie acciones o actividades en concreto, y deben cumplir con los requisitos que se plantearon inicialmente para poder cumplir con un sentido o lógica de ser añadidos.

4.2.3. CASOS DE USO

Las actividades que realizan los actores de un sistema de software son comúnmente conocidas como casos de uso. Se definen estos casos de uso describiendo un nombre de esta actividad, su identificador, que actor o actores se ven involucrados, qué eventos van a desencadenar cuando se realice la acción, los pasos que definen esta acción, que condiciones previas y posteriores se deben cumplir al desarrollo de esta acción, que suposiciones se tienen cuando se realiza la acción y qué requerimientos se cumplen.

Para definir estos casos de uso de manera comprensiva, se emplean formatos estandarizados que incluyen una serie de atributos esenciales [43]:

- **Nombre del caso de uso:** Es un título conciso y descriptivo que resume la actividad principal. Debe ser claro y orientarse a la meta que el actor busca lograr.
- **Identificador (ID):** Un código único que permite referenciar y rastrear el caso de uso a lo largo de todo el ciclo de vida del desarrollo de software.
- **Descripción:** Aclara, de manera breve y concisa, en qué consiste el caso de uso.
- **Actor o actores involucrados:** Se especifica quién o qué (persona u otro sistema) inicia el caso de uso y/o interactúa con él. Es importante tener en cuenta, también, que sirve para identificar a los actores primarios (quien inicia el caso de uso para lograr un objetivo) y, si los hay, a los actores secundarios (quienes participan, pero no inician el proceso).
- **Eventos desencadenantes:** Se describe el evento o la acción inicial que provoca el inicio del caso de uso. Puede ser una acción explícita del actor o un evento interno del sistema.
- **Pasos realizados:** Es la parte más importante del caso de uso ya que describe la secuencia lógica de interacciones entre el actor y el sistema.
- **Precondiciones:** Son las condiciones que deben ser verdaderas antes de que el caso de uso pueda comenzar. Si una precondición no se cumple, el caso de uso no puede iniciarse.
- **Postcondiciones:** Son las condiciones que deben ser verdaderas una vez que el caso de uso ha finalizado con éxito. Se detalla aquí el estado del sistema y del entorno después de que la actividad se haya completado.
- **Suposiciones:** Elementos que se asumen como ciertos o válidos durante el desarrollo del caso de uso, pero que podrían cambiar o ser inciertos en el futuro. Es importante documentarlas para identificar posibles riesgos o dependencias.
- **Requerimientos Funcionales y No Funcionales Asociados:** Aquí se referencian los requisitos funcionales que este caso de uso implementa directamente y los requisitos no funcionales relevantes que lo afectan.

A continuación, se detallan en qué consisten las acciones que realizan los actores de este proyecto.

Tabla 4.2—21 Caso de uso 1: Visualizar panel principal

| | | |
|--|---|-----------------------------------|
| Nombre del caso de uso | Visualizar panel principal | ID: C.U.1 |
| Actor(es) | Profesor | |
| Descripción | Permite al profesor ver, de manera gráfica, todas las posibilidades que tiene dentro del software. | |
| Eventos desencadenantes | El profesor va a la página principal de la web. | |
| Pasos realizados | | Información para los pasos |
| 1. Se inicia el programa | | Ninguna. |
| 2. Se inicia sesión como profesor | | Ninguna. |
| 3. Será visible la interfaz principal. | | Ninguna. |
| Precondiciones | El usuario debe haber creado una cuenta como profesor previamente antes de iniciar sesión. | |
| Postcondiciones | <p>* Si no existen sesiones grabadas, se indicará que no hay ninguna existente.</p> <p>* Si existen sesiones, se mostrarán de manera que se pueda seleccionar la que se quiera visualizar. Una vez seleccionada, se podrán ver detalles de gráficas que indiquen la atención de los alumnos, de manera global e individual.</p> | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-7, Administración de las sesiones; RF-8, Administración del perfil; RNF-1, Diseño estético, minimalista y sencillo; RNF-3, Código escalable; RNF-4, Código modular; RNF-5 Librerías de código abierto. | |
| Comentarios | Ninguno. | |

Tabla 4.2—22 Caso de uso 2: Iniciar la clase

| | | |
|--|--|---|
| Nombre del caso de uso | Iniciar la clase | ID: C.U.2 |
| Actor(es) | Profesor | |
| Descripción | Permite al profesor iniciar la sesión de la clase a realizar. | |
| Eventos desencadenantes | Se accede a la página inicial del profesor. | |
| Pasos realizados | | Información para los pasos |
| 1. Se rellenan los datos relevantes a la clase. | | Estos son “Nombre” y “Descripción” y se pueden encontrar en el menú principal del panel del profesor. |
| 2. Se pulsa el botón de “crear clase”. | | Ninguna |
| 3. Se comienza la clase. A partir de ese momento, cada alumno que entre en la clase será analizado. | | Ninguna. |
| 4. Durante los análisis, se muestran notificaciones relacionadas con la atención de los estudiantes. | | Se muestran solo las notificaciones de ausencias de atención y vigilia. |
| Precondiciones | El profesor debe haber iniciado sesión. | |
| Postcondiciones | <p>* Las notificaciones (en caso de aplicar) aparecerán por la pantalla del profesor.</p> <p>* Esto permitirá que el sistema registre los parpadeos, la apertura de los ojos, y las direcciones del foco de atención de los estudiantes para su posterior análisis.</p> | |
| Suposiciones | * El abandono de la aplicación en este paso se entiende como una cancelación del proceso de grabación. | |
| Requerimientos cumplidos | RF-1, Captura de la dirección de la cabeza; RF-2, Detección de parpadeos y ojos cerrados; RF-3, Medición del tiempo de la atención; RF-4, Medición del tiempo de somnolencia; RF-5, Grabación de las sesiones; RF-6, Sistema en tiempo real; RF-11, Notificación al profesor de la ausencia de atención; RF-12, Notificación al profesor de la somnolencia; RNF-1, Diseño estético, minimalista y sencillo; RNF-2, Anonimato de los alumnos; RNF-3, Código escalable; RNF-4, Código modular. | |
| Comentarios | Se muestra un código para que los estudiantes puedan entrar en la clase. | |

Tabla 4.2—23 Caso de uso 3: Parar grabación

| | | |
|--|---|-----------------------------------|
| Nombre del caso de uso | Terminar sesión | ID: C.U.3 |
| Actor(es) | Profesor | |
| Descripción | Permite al profesor detener completamente la clase iniciada y guardarla. | |
| Eventos desencadenantes | El profesor pulsa el botón para salir de la clase. | |
| Pasos realizados | | Información para los pasos |
| 1. Mientras se graba la clase, se pulsa un botón indicando la detención de la grabación. | | Ninguna. |
| Precondiciones | C.U.2 Iniciar grabación | |
| Postcondiciones | <ul style="list-style-type: none"> * C.U.10 Generar informe individual. * C.U.11 Generar informe global. * C.U.14 Salir de la clase. Se detendrá completamente la grabación y se almacenarán en la base de datos los detalles de la clase. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-5, Grabación de las sesiones; RF-6, Sistema en tiempo real; RNF-1, Diseño estético, minimalista y sencillo. | |
| Comentarios | Ninguno. | |



Tabla 4.2—24 Caso de uso 4: Visualizar sesión

| | | |
|---|--|-----------------------------------|
| Nombre del caso de uso | Visualizar sesión | ID: C.U.4 |
| Actor(es) | Profesor | |
| Descripción | Permite al profesor visualizar las grabaciones realizadas y ver los informes generados de manera individual y global. | |
| Eventos desencadenantes | El profesor se encuentra en su panel principal en la página. | |
| Pasos realizados | | Información para los pasos |
| 1. Se accede a la sección de las sesiones realizadas. | | Ninguna. |
| 2. Se selecciona una sesión. | | Ninguna. |
| 3. Se muestran los datos relevantes a esa sesión. | | Ninguna. |
| Precondiciones | Existen sesiones ya creadas. Si no existen, se muestra el aviso de que no hay. | |
| Postcondiciones | * Se muestra un seleccionable donde se puede elegir una de las clases que el profesor haya realizado. * La información mostrada será la individual (C.U.10) y la general (C.U.11) de los alumnos de la clase. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-3, Medición del tiempo de la atención; RF-4, Medición del tiempo de somnolencia; RF-5, Grabación de las sesiones; RF-7, Administración de las sesiones; RF-9, Resultados globales; RF-10, Resultados individuales; RNF-1, Diseño estético, minimalista y sencillo RNF-2, Anonimato de los alumnos; RNF-3, Código escalable; RNF-4, Código modular; RNF-5, Librerías de código abierto | |
| Comentarios | Ninguno. | |

Tabla 4.2—25 Caso de uso 5: Eliminar una sesión

| | | |
|---|---|-----------------------------------|
| Nombre del caso de uso | Eliminar una sesión | ID: C.U.5 |
| Actor(es) | Profesor | |
| Descripción | Permite al profesor borrar las sesiones realizadas de manera individual. | |
| Eventos desencadenantes | Se está en la sección de “sesiones” del panel del profesor. | |
| Pasos realizados | | Información para los pasos |
| 1. En la lista de sesiones almacenadas, cada sesión tiene un botón de borrar la sesión. Se pulsa. | | Ninguna. |
| 2. Se pide confirmación de la acción para evitar errores. | | Permite cancelar el caso de uso. |
| Precondiciones | Existen sesiones. | |
| Postcondiciones | La sesión se elimina de la base de datos. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-7, Administración de las sesiones; RNF-1, Diseño estético, minimalista y sencillo. | |
| Comentarios | Ninguno. | |

Tabla 4.2—26 Caso de uso 6: Modificar el perfil

| | | |
|--|--|---|
| Nombre del caso de uso | Modificar el perfil | ID: C.U.6 |
| Actor(es) | Profesor | |
| Descripción | Permite al profesor modificar datos como su nombre de usuario o su contraseña. | |
| Eventos desencadenantes | Se está en el apartado de “perfil” del panel del profesor activo. | |
| Pasos realizados | | Información para los pasos |
| 1. El profesor accederá al apartado de “Perfil” en el acceso identificado del sitio web. | | Ninguna. |
| 2. Se pulsa el botón de “Cambiar alias” o “Cambiar contraseña” | | No existen diferencias sustanciales entre ambas opciones. |
| 3. Se introducen los nuevos datos. | | Ninguna. |
| 4. Se pulsa guardar. | | Se pedirá confirmación de la acción. |
| Precondiciones | El profesor ha iniciado sesión. | |
| Postcondiciones | La información de la sesión se guarda en el sistema. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-8, Administración del perfil; RNF-1, Diseño estético, minimalista y sencillo; | |
| Comentarios | Ninguno. | |

Tabla 4.2—27 Caso de uso 7: Sostener la mirada

| | | |
|--|--|------------------|
| Nombre del caso de uso | Sostener la mirada | ID: C.U.7 |
| Actor(es) | Estudiante | |
| Descripción | El estudiante interactúa manteniendo la mirada en un lugar del espacio. | |
| Eventos desencadenantes | La cámara del estudiante está encendida. | |
| Pasos realizados | | |
| El estudiante actúa de manera natural frente a la cámara, y esto es capturado. | | |
| Precondiciones | <ul style="list-style-type: none"> * La sesión se está grabando. * El rostro del estudiante debe ser perceptible. | |
| Postcondiciones | La mirada se captura y se procesa en el sistema. Se captura la información relevante. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-1, Captura de la dirección de la cabeza; RF-3, Medición del tiempo de la atención; RF-5, Grabación de las sesiones; RF-6, Sistema en tiempo real; RF-7, Administración de las sesiones; RNF-3, Código escalable; RNF-4, Código modular; RNF-5, Librerías de código abierto. | |
| Comentarios | Se derivará en dos resultados posibles: atento o distraído. | |

Tabla 4.2—28 Caso de uso 8: Determinar una apertura de ojos y una frecuencia de parpadeo

| | | |
|--|--|------------------|
| Nombre del caso de uso | Determinar una apertura de ojos y una frecuencia de parpadeo | ID: C.U.8 |
| Actor(es) | Estudiante | |
| Descripción | El estudiante interactúa expresando una apertura de ojos y una frecuencia de parpadeo concretas. | |
| Evento desencadenado | Captura de la información relevante. | |
| Pasos realizados | | |
| El estudiante actúa de manera natural frente a la cámara, y esto es capturado. | | |
| Precondiciones | <ul style="list-style-type: none"> * La sesión se está grabando. * El rostro del estudiante debe ser perceptible. | |
| Postcondiciones | La apertura de ojos y la frecuencia de parpadeo se capturan y se procesan en el sistema. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-2, Detección de parpadeos y ojos cerrados; RF-3, Medición del tiempo de la atención; RF-5, Grabación de las sesiones; RF-6, Sistema en tiempo real; RF-7, Administración de las sesiones; RNF-3, Código escalable; RNF-4, Código modular; RNF-5, Librerías de código abierto. | |
| Comentarios | Se derivará en dos resultados posibles: somnoliento o despierto. | |

Tabla 4.2—29 Caso de uso 9: Determinar nivel de atención

| | | |
|---|--|---|
| Nombre del caso de uso | Determinar atención y somnolencia | ID: C.U.9 |
| Actor(es) | Inteligencia Artificial | |
| Descripción | La Inteligencia Artificial indica si un estudiante está atento y/o durmiéndose, basándose en los inputs que éste genera. | |
| Evento desencadenado | Generación de los datos de información principal. | |
| Pasos realizados | | Información para los pasos |
| 1. Se captura la imagen de la cara grabada en tiempo real. | | Ninguna. |
| 2. Se determina el ángulo de la cabeza. | | Ninguna. |
| 3. Se compara el ángulo con los ángulos pasados para encontrar qué variaciones existen. | | Ninguna. |
| 4. Se determina si el ángulo se encuentra dentro de un margen considerado como aceptable, se considerará “atento” o no. | | Ninguna. |
| 5. Se captura la apertura de los ojos. | | Ninguna. |
| 6. Se mide la velocidad y la frecuencia de parpadeo cada segundo. | | La velocidad de parpadeo diferenciará entre si la persona está pestañeando o con los ojos cerrados, y la frecuencia determinará si está despierto o no. |
| 7. Se determina si la persona está durmiendo o no. | | Ninguna. |
| 8. Si hay un cambio de estado en la atención o la persona se duerme, se notifica en el momento. | | Esta lógica va independiente del envío de la información. |
| 9. Se envía un resumen de la atención cada minuto. | | Este resumen se guarda en la base de datos con información de si se presta atención o no, si se ha dormido o no, por cuánto tiempo en ambos casos, qué usuario ha enviado este resumen y en qué minuto lo hace. |
| Precondiciones | Se han realizado los casos de uso 2, 7 y 8. | |
| Postcondiciones | La información de esta captura de la atención se almacenará de modo que se indicará el usuario que la ha generado, y el momento en el que lo hizo. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-1, Captura de la dirección de la cabeza; RF-2, Detección de parpadeos y ojos cerrados; RF-3, Medición del tiempo de somnolencia; RF-4, Reconocimiento de la postura; RF-5, Grabación de las sesiones; RF-6, Sistema en tiempo real; RF-10, Resultados individuales; RF-11, Notificación al profesor de la ausencia de la atención; RF-12, Notificación al profesor de somnolencia; RNF-2, Anonimato de los alumnos; RNF-3, Código | |

| | |
|--------------------|--|
| | escalable; RNF-4, Código modular; RNF-5, Librerías de código abierto |
| Comentarios | Esta operación se realiza del lado del cliente. |

Tabla 4.2—30 Caso de uso 10: Generar informe individual

| | | |
|---------------------------------|--|--|
| Nombre del caso de uso | Generar informe individual | ID: C.U.10 |
| Actor(es) | Inteligencia Artificial | |
| Descripción | La Inteligencia Artificial recopila toda la información de un estudiante para generar un análisis de su atención. | |
| Evento desencadenado | Generación de los datos de información principal. | |
| | Pasos realizados | Información para los pasos |
| | 1. Se recuperan los datos pertenecientes al C.U. 9, “Determinar atención y somnolencia” en cada marca temporal de la grabación | Ninguna. |
| | 2. Se agrupan esos datos de maneras distintas para mostrar las diferentes gráficas. | Estas agrupaciones contendrán marcas de tiempo con las dos variables de atención y somnolencia mostradas de maneras distintas. |
| | 3. Se estima cuánto porcentaje del tiempo total de la clase ha estado atento y despierto el estudiante. | Ninguna. |
| Precondiciones | Se determinado la atención y somnolencia de los alumnos y, por tanto, se ha realizado el C.U.9. | |
| Postcondiciones | Ninguna. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-10, Resultados individuales; RNF-2, Anonimato de los alumnos, RNF-3, Código escalable; RNF-4, Código modular. | |
| Comentarios | OPERACIÓN REALIZADA DEL LADO DEL PROFESOR. | |

Tabla 4.2—31 Caso de uso 11: Generar informe global

| | | |
|---------------------------------|---|--|
| Nombre del caso de uso | Generar informe global | ID: C.U.11 |
| Actor(es) | Inteligencia Artificial | |
| Descripción | La Inteligencia Artificial recopila toda la información de todos los estudiantes de una sesión para generar un análisis de su atención. | |
| Evento desencadenado | Generación de los datos de información principal. | |
| | Pasos realizados | Información para los pasos |
| | 1. Se recuperan los datos pertenecientes al C.U.10, “Generar informe individual”. | Ninguna. |
| | 2. Se agrupan esos datos de maneras distintas para mostrar las diferentes gráficas. | Estas agrupaciones contendrán marcas de tiempo con las dos variables de atención y somnolencia mostradas de maneras distintas. |
| | 3. Se estima cuánto porcentaje de la clase ha estado atenta en la totalidad de la clase. | Ninguna. |
| Precondiciones | Se han generado los informes individuales y, por tanto, se ha hecho el C.U.9. | |
| Postcondiciones | Ninguna. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RF-9, Resultados globales; RNF-2, Anonimato de los alumnos; RNF-3, Código escalable; RNF-4, Código modular. | |
| Comentarios | OPERACIÓN REALIZADA DEL LADO DEL PROFESOR. | |

Tabla 4.2—32 Caso de uso 12: Mostrar notificaciones

| | | |
|---|--|--|
| Nombre del caso de uso | Mostrar notificaciones | ID: C.U.12 |
| Actor(es) | Inteligencia Artificial | |
| Descripción | La Inteligencia Artificial, al detectar alumnos distraídos, mandará una notificación al profesor para notificarlo. | |
| Evento desencadenado | Aviso al profesor de la distracción. | |
| Pasos realizados | | Información para los pasos |
| 1. Se detecta un cambio en la atención o una señal de somnolencia por parte de un alumno. | | Ninguna. |
| 2. Se activa una notificación que se envía al servidor. | | Ninguna. |
| 3. El servidor le envía esta señal al profesor. | | Esta señal contendrá el nombre del alumno que no está atento y el momento en el que ha ocurrido. |
| Precondiciones | Se ha realizado el C.U.9: Determinar nivel de atención | |
| Postcondiciones | Ninguna. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RNF-1, Diseño estético, minimalista y sencillo; RF-11, Notificación al profesor de la ausencia de atención; RF-12, Notificación al profesor de somnolencia; RNF-3, Código escalable; RNF-4, Código modular; RNF-5, Librerías de código abierto | |
| Comentarios | OPERACIÓN REALIZADA DEL LADO DEL PROFESOR. | |

Tabla 4.2—33 Caso de uso 13: Asistir a la clase

| | | |
|--|---|--|
| Nombre del caso de uso | Asistir a la clase | ID: C.U.13 |
| Actor(es) | Profesor, Estudiante | |
| Descripción | Un profesor puede empezar una clase a la que asistirá automáticamente, y sus alumnos pueden ingresar a ella. | |
| Evento desencadenado | Ninguno. | |
| Pasos realizados | | Información para los pasos |
| 1. Desde el inicio de la web, se selecciona la vista de estudiante o la vista de profesor. | | Por defecto se verá la vista del estudiante. |
| 2. Si se es profesor, se inicia sesión y se podrá crear una clase. | | Iniciar la clase, Caso de Uso 2. |
| 3. Si se es estudiante, se ingresa el alias del profesor para asistir. | | Ninguno. |
| Precondiciones | Ninguna. | |
| Postcondiciones | Ninguna. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RNF-1, Diseño estético, minimalista y sencillo; RF-2, Anonimato de los alumnos; RNF-4, Código modular; RNF-5, Librerías de código abierto | |
| Comentarios | Ninguno. | |

Tabla 4.2—34 Caso de uso 14: Salir de la clase

| | | |
|-------------------------------------|---|-----------------------------------|
| Nombre del caso de uso | Salir de la clase | ID: C.U.14 |
| Actor(es) | Estudiante, Profesor | |
| Descripción | Un estudiante o un profesor puede abandonar una clase a la que está asistiendo. | |
| Evento desencadenado | Ninguno. | |
| Pasos realizados | | Información para los pasos |
| 1. Se pulsa el botón de “Abandonar” | | Ninguna. |
| Precondiciones | Se ha realizado el C.U.13: Asistir a la clase | |
| Postcondiciones | Los alumnos salen de la clase y mandan el resumen de su atención de ese minuto. | |
| Suposiciones | Ninguna. | |
| Requerimientos cumplidos | RNF-1, Diseño estético, minimalista y sencillo; | |
| Comentarios | Al salir el profesor, todos los estudiantes saldrán también. | |

4.3. DISEÑO

Para entender mejor qué tipo de actividades, implementar correctamente el sistema y poder distinguir correctamente cada componente, se diseña el software que se va a realizar. Para diseñarlo, se pueden definir sus interacciones, responsabilidades, partes y relaciones que hay en él.

DIAGRAMAS UML

Resulta de utilidad representar el sistema de manera gráfica por medio de diagramas. En el contexto del desarrollo software, se utilizan diagramas UML (*Unified Model Language*, por sus siglas en inglés, Lenguaje Unificado de Modelado en español) [44].

Un diagrama UML es una forma de visualizar sistemas y software utilizando un lenguaje de iconografías, palabras y gráficos común con el fin de comprender el sistema, la arquitectura del código y su implementación.

Los diagramas UML facilitan la comprensión de las ideas y sistemas que resultan complejos, es por ello por lo que se emplean para el desarrollo del software.

A continuación, se describe detalladamente el diseño del sistema con sus diagramas fundamentales para los conceptos más importantes.

4.3.1. DIAGRAMA DE CASOS DE USO

En la Figura 4.3—1 Diagrama de casos de uso, se representa gráficamente cómo cada uno de ellos participa en el funcionamiento general de la aplicación mediante un diagrama de casos de uso. Este diagrama representa, también, qué relaciones tienen los actores entre sí.

En este diagrama, también, es posible identificar qué casos de uso dependen de otros, por ejemplo, para salir de clase primero necesitas haber asistido; y detectar la atención y la somnolencia dependen de los casos de uso de sostener la mirada y determinar una apertura de ojos y una frecuencia de parpadeo.



Figura 4.3—1 Diagrama de casos de uso

4.3.2. DIAGRAMA DE ESTADOS

Un caso de uso, al describir una funcionalidad específica del sistema desde la perspectiva del usuario, implica una serie de situaciones clave o estados por los que el sistema puede transitar para cumplir con ese objetivo. En un diagrama de estados (o diagrama de máquina de estados), se pretende, precisamente, reflejar de manera formal y visual todas las situaciones posibles en las que se puede encontrar una parte del software o el sistema a lo largo de la ejecución de un caso de uso o de un proceso particular.

Este tipo de diagrama es importante porque define claramente el comportamiento dinámico del sistema. No solo muestra los distintos estados por los que puede pasar el software, sino que, además, especifica qué condiciones o eventos deben cumplirse para que se produzca un cambio de un estado a otro.

En este caso, se analiza el flujo de estados y el procedimiento del caso de uso del análisis de la atención de un alumno.

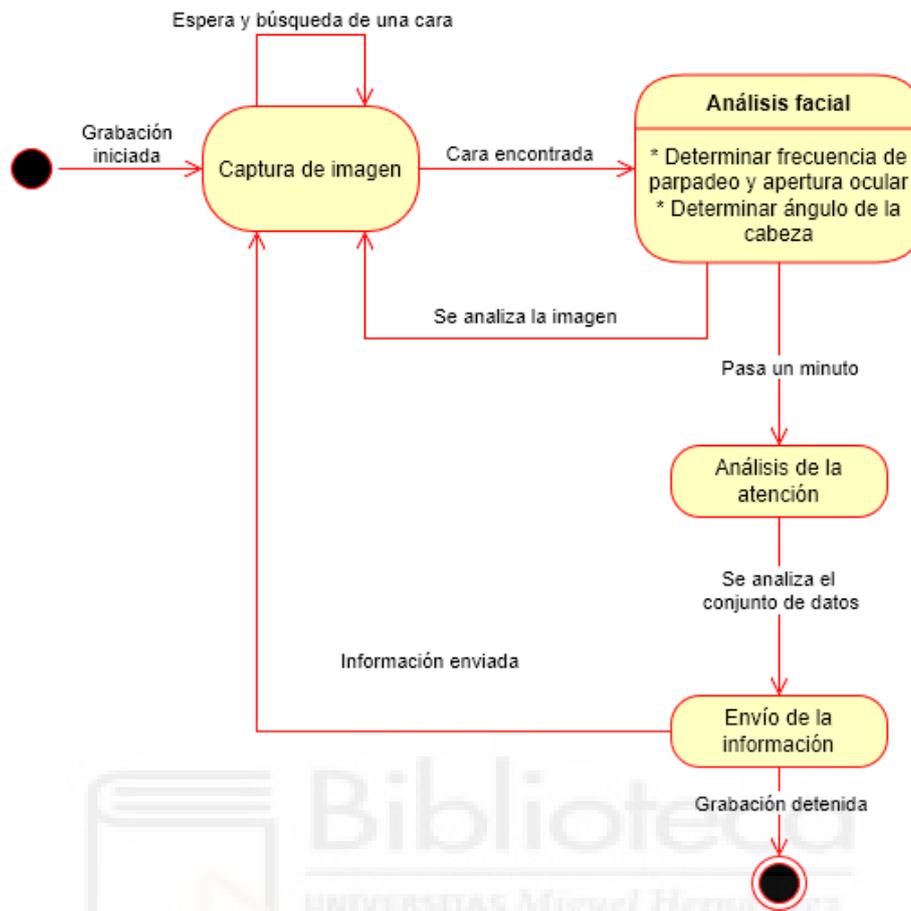


Figura 4.3—2 Diagrama de estado

Este diagrama refleja los siguientes estados:

- **Captura de imagen:** el sistema se inicia al entrar a una clase y espera a encontrar una cara. Si encuentra más de una, capturará únicamente la primera que encuentre. Se mantendrá en este estado hasta que encuentre el rostro.
- **Análisis facial:** si el estudiante es encontrado, se le analiza la posición que tiene en ese momento su cabeza, así como también la apertura de los ojos. Almacenará la información recopilada en un arreglo.
- **Análisis de la atención:** Se reúnen todos los datos recogidos en el periodo y se analizará la atención que ha tenido el estudiante.
- **Envío de la información:** se envía la información al profesor y se vuelve a intentar capturar la imagen. Si se detiene la clase, bien sea por abandono o porque se cierra la clase, se intentará enviar la información obtenida aunque esté incompleta.

4.3.3. DIAGRAMA DE SECUENCIA

Un diagrama de secuencia es útil para poder indicar cómo se comportan los diferentes elementos del sistema entre sí a lo largo del ciclo de vida del programa. Se especificará cómo funciona el proceso de grabación desde esta perspectiva para poder comprender mejor cómo se relacionan los elementos entre sí.

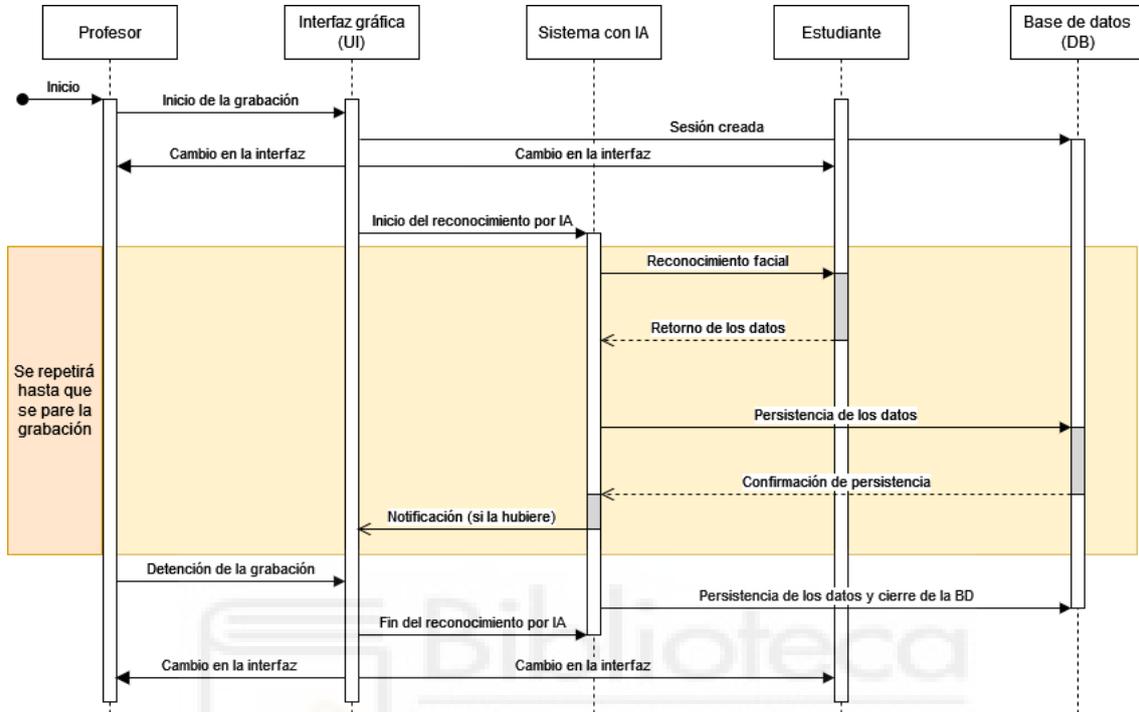


Figura 4.3—3 Diagrama de secuencia

En la Figura 4.3—3 Diagrama de secuencia se pueden observar cinco elementos:

- **Profesor**, que realiza acciones de inicio y detención de las clases, así como de recibir información por pantalla.
- **Interfaz gráfica**, que recibe eventos y cambia su visualización para el profesor y para el estudiante.
- **Sistema con IA**, que realiza operaciones fundamentales de captura, manejo, análisis y envío de los datos.
- **Estudiante**, que recibe cambios en la interfaz y genera los datos fundamentales que necesita la aplicación.
- **Base de datos**, que recibe información del resto de elementos y realiza el papel de banco de datos.

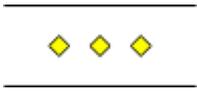
Este diagrama muestra la línea de vida que tienen estos elementos y los eventos que ocurren.

4.3.4. DIAGRAMA DE FLUJO

El diagrama de flujo sirve para conocer mejor cómo se comportará el sistema dependiendo de las circunstancias que se encuentre en su ciclo de vida. En este caso, se analiza la casuística del proceso de grabación sin tener en cuenta su pausa ni detención.

A pesar de que existen más iconos para representar otro tipo de conceptos, en la **¡Error! No se encuentra el origen de la referencia.** se puede observar todo el procedimiento con una serie de iconografías que representan lo siguiente:

Tabla 4.3—4 Iconografía del diagrama de flujo

| Icono | Descripción |
|---|---|
|  | Describe un proceso que hace de nodo y que, al terminar, puede llevar a otro proceso. |
|  | Es un nodo que define una posibilidad de bifurcar el flujo de procedimiento de acciones. |
|  | Se realiza una acción en la que intervienen entradas o salidas de los datos en el sistema o desde el sistema. |
|  | Es la entidad que hace el papel de base de datos. Se define para indicar que se almacenan de la base de datos o que salen de ahí. |
|  | Indica que el flujo de procesos se paraleliza en varios procesos o que se une en uno secuencial. |

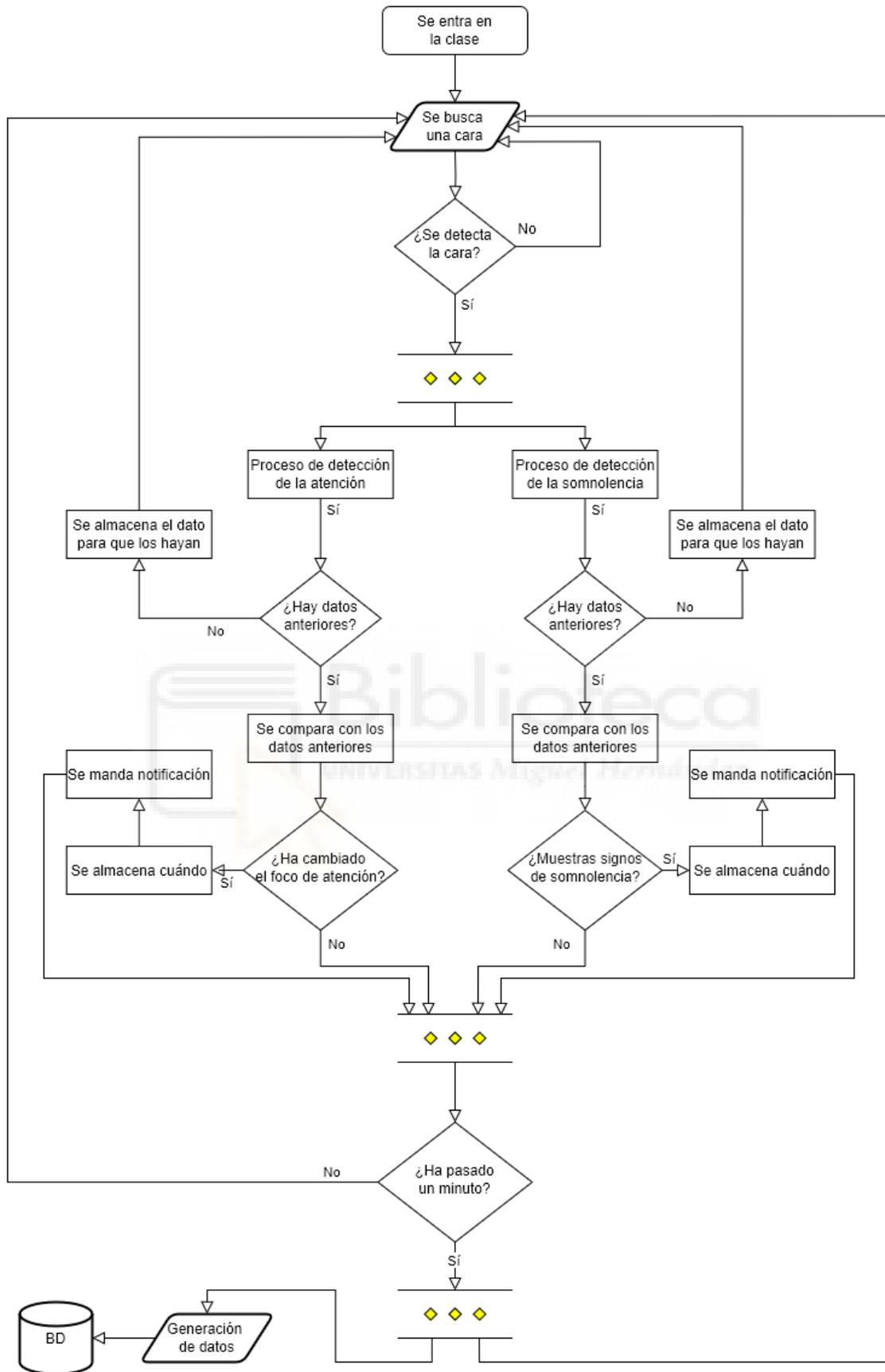


Figura 4.3—5 Diagrama de flujo

4.3.5. DIAGRAMA DE CLASE

Un diagrama de clase nos sirve para comprender mejor qué objetos componen nuestro sistema y de qué manera se relacionan entre sí. Esto es especialmente útil para identificar, también, los componentes de la base de datos.

Este diagrama genérico ayuda a comprender mejor qué información es relevante para nuestro sistema.

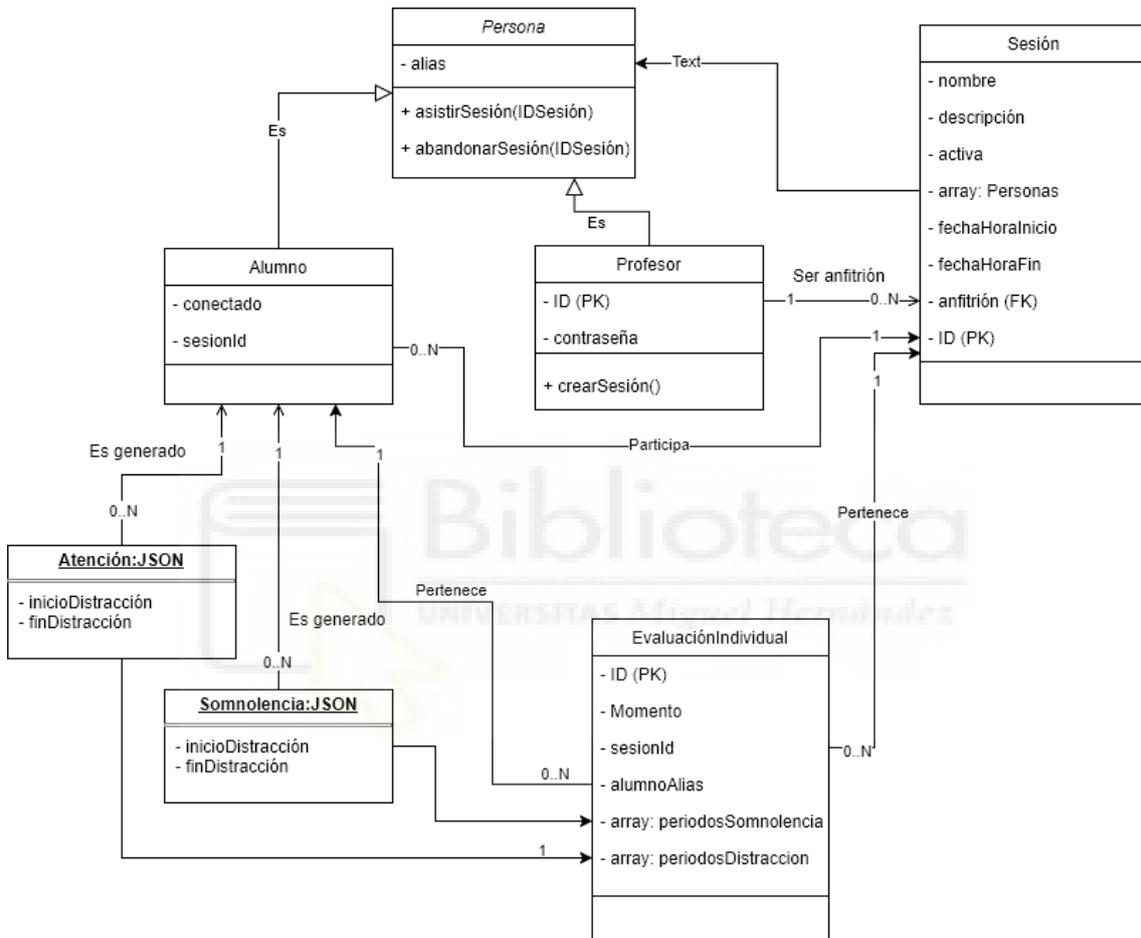


Figura 4.3—6 Diagrama de clase

4.3.6. DIAGRAMA DE BASE DE DATOS

Este diagrama sirve para representar los elementos que se introducirán en la base de datos, así como sus atributos.

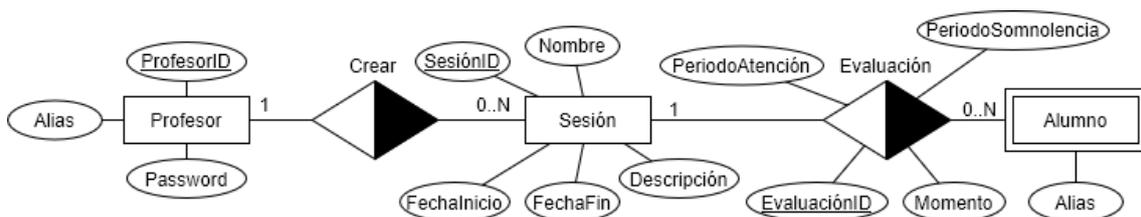


Figura 4.3—7 Diagrama de base de datos

A continuación, se detallan los atributos de cada tabla elemento dentro de la base de datos:

Tabla 4.3—8 Columnas de la tabla de Profesor

| Columnas de la tabla de Profesor | | |
|----------------------------------|----------------------|--|
| Nombre | Tipo | Descripción |
| ProfesorID | Entero | (PK) Identificador del usuario. |
| Alias | Cadena de caracteres | Nombre que aparecerá en pantalla. |
| Password | Cadena de caracteres | No se contemplará que sea una contraseña segura. |

Tabla 4.3—9 Columnas de la tabla de Alumno

| Columnas de la tabla de Alumno | | |
|--------------------------------|----------------------|--|
| Nombre | Tipo | Descripción |
| Alias | Cadena de caracteres | (PK1) Nombre que aparecerá en pantalla. |
| SesionID | Entero | (FK, Sesión) (PK2) Es el identificador de la clase a la que asiste el alumno. |
| Conectado | Booleano | Define si el usuario está conectado a la sesión o no |

Tabla 4.3—10 Columnas de la tabla de Sesión

| Columnas de la tabla de Sesión | | |
|--------------------------------|----------------------|---|
| Nombre | Tipo | Descripción |
| SesionID | Entero | (PK) Identificador de la sesión. |
| ProfesorID | Entero | (FK, Profesor) Es el identificador del profesor que da la clase. |
| Nombre | Cadena de caracteres | Nombre que aparecerá en pantalla. |
| Descripción | Cadena de caracteres | Descripción breve de no más de 200 caracteres. |
| FechaInicio | Fecha | Describe cuándo ha empezado la clase. |
| FechaFin | Fecha | Describe cuándo ha acabado la clase. |

Tabla 4.3—11 Columnas de la tabla de Evaluación

| Columnas de la tabla de Evaluación | | |
|------------------------------------|--------|--|
| Nombre | Tipo | Descripción |
| EvaluacionID | Entero | (PK) Identifica la evaluación |
| Momento | Tiempo | (UK) Momento temporal. |
| SesionID | Entero | (FK, Sesión) Identificador de la clase. |
| Alumno | Entero | (FK, Alumno) Es el alias del alumno que asiste a la clase. |
| periodoDistracción | JSON | Archivo JSON que indica todos los periodos en los que ha habido cambios de atención a lo largo de un minuto determinado. |
| periodoSomnolencia | JSON | Archivo JSON que indica todos los periodos en los que ha habido signos de somnolencia a lo largo de un minuto determinado. |

4.3.7. INFORMES DE LOS DATOS CAPTURADOS

En este informe se pretende determinar y explicar los datos capturados de los rostros de los estudiantes para determinar su nivel de atención durante las clases. Para capturar estos datos, como ya se ha mencionado, se han utilizado las soluciones de Mediapipe.

4.3.7.1. DATOS CLAVE A DETECTAR

Se identifican puntos específicos en el rostro, como ojos, cejas, boca y nariz haciendo uso de Face Mesh de Mediapipe. Se captura información como:

- **Detección de la dirección de la mirada (*Eye-Tracking*).**

La captura de la mirada determina hacia dónde está mirando el alumno. Los parámetros que determinan hacia dónde se está dirigiendo la mirada son la posición de los ojos y la nariz para detectar la dirección a la que se está enfocando la atención. Es importante para indicar si el alumno está prestando atención a la clase, ya que, como se ha mencionado previamente, la posición de la cabeza y la mirada constituye el 88.7% de la atención.

- **Parpadeo (*Blink detection*)**

Se detecta la frecuencia de parpadeo y la duración de los ojos cerrados haciendo uso de la ratio de apertura ocular (EAR, *Eye Aspect Ratio*). Es importante ya que los parpadeos excesivos o prolongados pueden indicar fatiga o falta de atención.

Como se ha podido observar en puntos anteriores, la velocidad de cierre del párpado debe ser tal que permita que una persona parpadee entre 15 y 20 veces por minuto. Valores por encima de 20 se consideran como indicios de somnolencia.

Lo mismo ocurre con el retorno a la posición abierta de los párpados: si es de 1-2 segundos se asocian con un estado de "Somnolencia", durante 2-3 segundos con un estado de "Muy somnoliento" y durante más de 4 segundos "extremadamente somnoliento". Por simplificar tanto el proceso como el análisis de la atención, así como también sus innecesarios costes computacionales, y también para ofrecer flexibilidad en el contexto del uso, se ha reducido a detectar si hay somnolencia o no.

4.3.7.2. CONSIDERACIONES TÉCNICAS DE CAPTURA Y PROCESAMIENTO DE DATOS

Este apartado profundiza en los aspectos técnicos claves relacionados con la captura, el procesamiento y la interpretación de los datos utilizados para la detección de la atención y la somnolencia en el sistema.

FRECUENCIA DE MUESTREO DE LOS DATOS

La naturaleza en tiempo real de los requisitos del sistema para la monitorización de la atención exige una captura de datos continua y a una alta frecuencia. Por ello, conviene tomar muestras a 30 imágenes por segundo. Esta tasa de cuadros está cerca del estándar de 24 FPS de medios como la televisión o cine, y en aplicaciones de vídeo proporciona una resolución temporal adecuada para capturar movimientos oculares rápidos, parpadeos y cambios en la orientación de la cabeza, elementos cruciales para los algoritmos de detección [45].

Cada segundo, se procesan los datos de estas 30 imágenes para extraer las métricas relevantes (ej., dirección de la mirada, frecuencia de parpadeo). Cada minuto, se analiza desde el lado del cliente la atención del usuario, y se envía al servidor un resumen de las métricas capturadas. Este resumen contiene información sobre:

- **Periodos de cambio en la dirección del foco de atención:** se registra cuándo y por cuánto tiempo el usuario desvió su mirada de la zona de interés.
- **Periodos de somnolencia:** al igual que con la atención, se cuantifica el número de veces que se detectaron signos de somnolencia, junto con las marcas de tiempo específicas de cada momento.

Esta estrategia de muestreo de los datos minimiza la carga de red al no tener que transmitir datos sin procesar de cada *frame*, a la vez que permite reflejar de manera detallada la evolución del estado de atención y vigilia del alumno.

ESTRATEGIA DE ALMACENAMIENTO

Para este proyecto se ha adoptado una estrategia centralizada en la que todos los datos generados persisten en la base de datos del servidor de manera permanente. Esto asegura que la información esté disponible siempre para futuras consultas por parte del profesor.

Como se ha podido observar en apartados anteriores, por diseño, del lado del cliente no se almacena esta información capturada. Esta decisión permite evitar el consumo excesivo de almacenamiento y memoria del dispositivo del usuario cliente, y cumple las normativas de privacidad de datos ya que solo se envía y almacena en el servidor el resumen analítico creado por un usuario.

DEFINICIÓN DE UMBRALES PARA LA DETECCIÓN DE ATENCIÓN Y SOMNOLENCIA

Para clasificar de manera efectiva el estado de atención y somnolencia del usuario, se han establecido una serie de umbrales críticos basados en las métricas de *eye tracking* y *blink detection*. Estos umbrales son fundamentales para la lógica interna del sistema que determina si una persona está atenta y despierta, o si muestra signos de distracción o somnolencia

Detección de Atención (Eye Tracking)

La atención se evalúa principalmente a través del análisis de la dirección de la mirada del usuario. Se considera que el usuario está atento si su mirada se mantiene consistentemente dentro de una zona de interés durante la mayor parte del tiempo. Por otro lado, se interpreta como distraído cuando la dirección de la mirada del usuario se desvía significativamente de la zona de interés.

Para cuantificar este cambio de foco, el sistema mantiene un histórico de las direcciones de mirada del estudiante durante un periodo de tiempo determinado. Para este proyecto, se han establecido los últimos 15 segundos. Después, se calcula una media de las direcciones de mirada dentro de ese histórico. Si la diferencia angular (en grados) entre la dirección de la mirada actual y el promedio histórico es superior a un umbral que en este proyecto se ha definido por 60° , se considera que ha habido un cambio de foco de atención. Si se mantiene esta desviación, se puede confirmar un estado de distracción.

El umbral de 30° por cada lado desde el punto que se considera como “**foco de atención**” y se ha elegido en base a los resultados del trabajo de M. Raca et al [13]. En cuanto al criterio de considerar los últimos 15 segundos para la media, ha sido por decisión propia por ofrecer buenos resultados en este proyecto.

Si bien la teoría ha aportado la gran parte de los fundamentos del proyecto, es interesante observar que la experimentación y las pruebas han resultado ser una parte importante dentro del proyecto. Debido a que la naturaleza del sistema es la de ser flexible y adaptable a cualquier tipo de contexto, la experimentación con los diferentes parámetros del análisis facial ha servido para ajustar mejor los umbrales de detección de la atención.

Detección de Somnolencia (Blink Detection)

La somnolencia se evalúa a través de la frecuencia y patrón del parpadeo, una métrica reconocida en estudios de factores humanos, como se ha podido observar durante el punto anterior “Introducción al estado de somnolencia”.

Se considera que la persona está despierta cuando tiene un parpadeo con una frecuencia normal, de entre 15 y 20 veces por minuto. Si se parpadea menos de ese rango, tampoco se considera que el usuario esté en un estado de somnolencia.

Por otro lado, la somnolencia se detecta si un parpadeo excede una duración normal (es decir, el ojo permanece cerrado por un periodo de tiempo prolongado) es un fuerte indicador de somnolencia. Además, si la frecuencia de parpadeo se desvía significativamente de la normalidad (por encima de 20 parpadeos por minuto), puede ser un signo de somnolencia.

4.3.8. DISEÑO DE LA INTERFAZ

La interfaz de usuario (UI) de este sistema ha seguido principios fundamentales de usabilidad y experiencia de usuario (UX), buscando un equilibrio entre simplicidad, intuición y completitud funcional. El objetivo principal es que los usuarios, ya sean estudiantes o profesores, puedan interactuar con la aplicación de forma eficiente y sin esfuerzo, accediendo a la información esencial de manera rápida y clara.

En este apartado se detallan los aspectos más importantes del diseño de la interfaz.

4.3.8.1. CARACTERÍSTICAS DE LA INTERFAZ

El aspecto de la interfaz se ha diseñado con el propósito de ser minimalista, con el objetivo de evitar elementos redundantes o distractores para mantener el foco en la información relevante. Este objetivo es importante de cumplir, ya que todo el objetivo del sistema es de analizar la atención del alumnado de una clase, y una interfaz llamativa, además de sesgar los datos porque provocaría distracciones, no sería coherente a los principios del proyecto.

4.3.8.2. BOCETOS DE LA INTERFAZ

Antes de desarrollar de manera definitiva la interfaz de usuario del sistema, es interesante bocetar cómo se verá para poder detectar cuáles son los puntos con más peso. Por tanto, resulta de utilidad hacer un boceto ya que sirve de guía sencilla para conseguir el objetivo que se pretende conseguir.

Conforme se avanza en el proyecto, es necesario redefinir no solo el aspecto del software si no también sus funcionalidades. Es por ello por lo que las ideas iniciales del diseño de la interfaz tienen un diseño tan diferente a como se hizo finalmente. Tras varias iteraciones en el diseño, se acabó llegando a un diseño todavía más estético y minimalista.

INTERFAZ DEL PORTAL DE INICIO

La interfaz del portal de inicio es la más importante, ya que es aquella en la que se espera que llegue el usuario por primera vez cuando visita una página web. En el caso de este proyecto, en esta ventana se mostrará al usuario una serie de elementos gráficos que le permitirán elegir si es un estudiante o un profesor.

Figura 4.3—12 Boceto del inicio de sesión, estudiante seleccionado

Figura 4.3—13 Boceto del inicio de sesión, profesor seleccionado

Se puede observar que en el primer bocetado de la interfaz se mostraron los elementos de formulario de inicio de sesión a un lado y dos botones que permitirían cambiar la interfaz según escojas un rol u otro. Se ve que la ventana principal muestra, lo primero, el inicio de sesión para hacer uso de la aplicación. Esto es fundamental, ya que es una aplicación cuyo propósito es realizar una tarea en la que se depende de la sesión de un usuario, y su intención no es la de mostrar contenido.

En la venta de inicio de sesión, el usuario podrá elegir de manera sencilla qué rol ocupa. Si es un estudiante, no se guardará información más allá que la de su nombre de usuario. Este estudiante podrá unirse a una clase que cree un profesor. Si es un profesor, se pedirá la contraseña, así como también se podrá dar de alta en la plataforma.

ACCESO IDENTIFICADO DEL PROFESOR

Una vez el profesor haya iniciado la sesión, se mostrará una ventana donde tendrá acceso a todas las clases que ha realizado. En un principio, se pensó en mostrar un identificador único de la sesión para poder llevar un mejor control, así como también algunos de los detalles que conforman la clase. Desde esta interfaz se pretendía poder modificar el nombre y descripción de la clase, o también eliminar la sesión.



Figura 4.3—14 Boceto de la vista de inicio del profesor

Esta interfaz permite que, si se pulsa en el botón de la papelera, se elimine la clase de manera permanente, junto con una previa confirmación del usuario para evitar errores accidentales.

También se puede observar la posibilidad de editar una clase haciendo uso del botón del lápiz. Aunque inicialmente se contempló el desarrollo completo del caso de uso relacionado con la edición de una clase, este no fue finalmente implementado en la versión final del proyecto debido a limitaciones de tiempo y a la priorización de otras

funcionalidades consideradas más críticas para los objetivos del sistema. Aun así, se dejó preparado el diseño visual y la arquitectura base del *backend* para una posible integración futura.

Por otro lado, si el usuario hace clic en el recuadro de una sesión, podrá ver sus detalles. Este apartado fue maquetado debido a que suponía una complejidad mayor que debía ser representada visualmente:

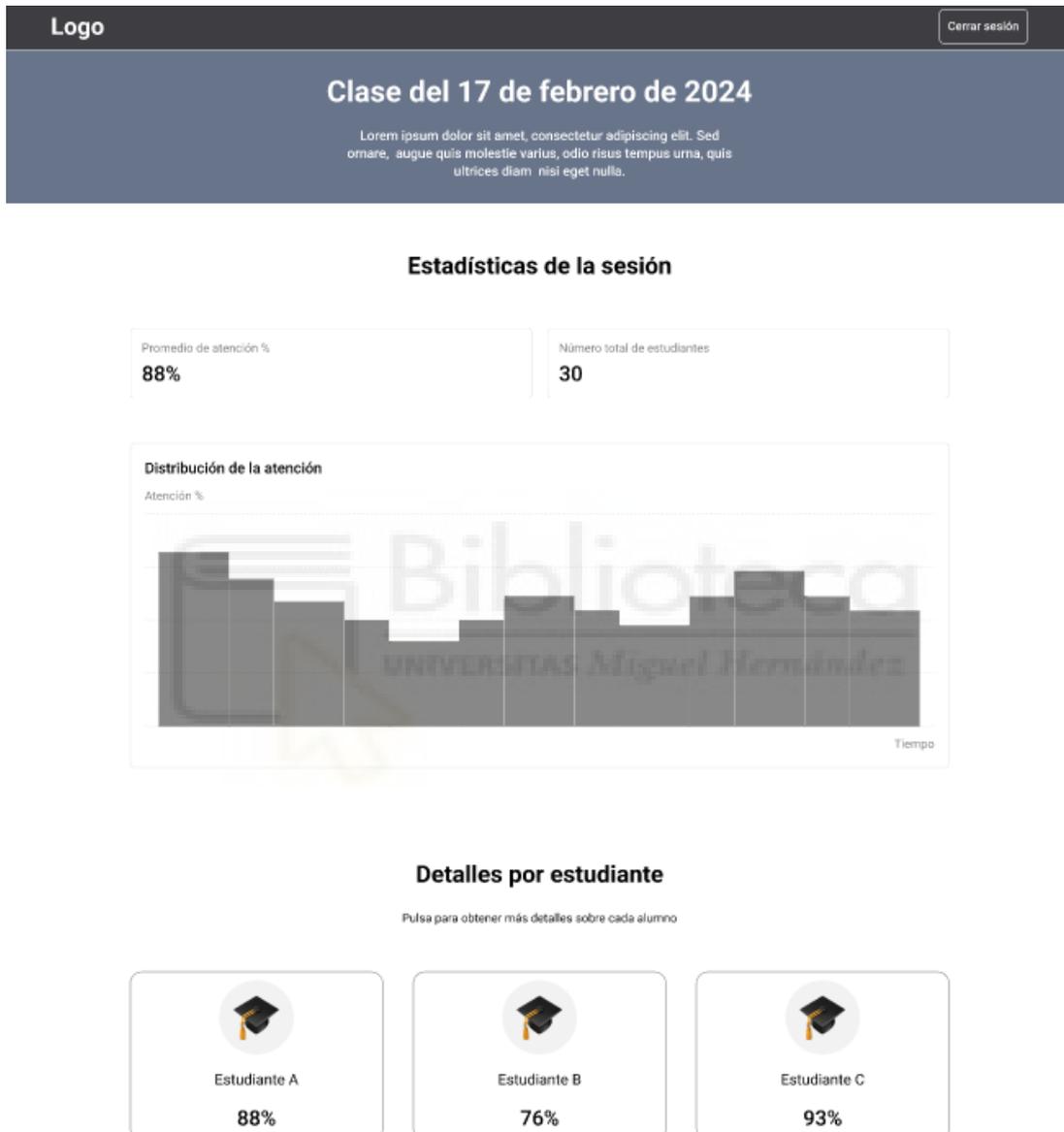


Figura 4.3—15 Boceto de la vista de las estadísticas de una clase

Desde esta ventana, el profesor puede visualizar en profundidad las estadísticas recogidas durante una de sus sesiones para, así, obtener conclusiones relevantes sobre el desempeño de la clase. Esta sección está pensada para mostrar un resumen global e intuitivo, en el que se representa gráficamente la atención media de los alumnos en distintos intervalos de tiempo, permitiendo identificar con facilidad en qué momentos se ha mantenido la atención alta y cuándo ha descendido. Para ello, se hace uso de gráficos de líneas y barras generados en base a los datos almacenados durante la clase.

4.4. ESTUDIO DE LA EVOLUCIÓN DEL PROYECTO

Como cualquier proyecto de desarrollo de software, los requerimientos del software se debieron ajustar según se fue avanzando en su desarrollo. La idea inicial de este proyecto ha ido variando e iterando según se fue aprendiendo sobre las diferentes tecnologías y el contexto de caso de uso; así como también se fue aprendiendo sobre qué era más viable tanto computacionalmente como en su escalabilidad en otros ámbitos.

El software se diseñó de modo que se pudiese servir como base para, en un futuro, ampliar con la detección de más características faciales, más gestión de los usuarios, mejores algoritmos y demás funcionalidades. Ello se logró gracias a la modularidad de los componentes del código y a la selección minuciosa de las tecnologías.

En este apartado se desarrollan los motivos por los que se avanzó en una dirección a la hora de realizar este proyecto, qué otras alternativas se consideraron y por qué fueron rechazadas finalmente.

4.4.1. IDEA INICIAL Y PRIMERAS ETAPAS

La idea inicial era, únicamente, desarrollar una aplicación que detectase la atención de los estudiantes en tiempo real. Primero, se optó por diseñar un sistema que se ejecutase únicamente de manera local en el ordenador del profesor y, por el cual, se emplearía una única cámara que se pondría en un aula. Esta cámara sería la Intel Realsense D435, la cual tiene un gran campo de visión y tiene la capacidad de capturar imágenes en 3D gracias a su sensor de profundidad [46].



Figura 4.4—1 Cámara Intel Realsense D435

La ventaja de este dispositivo era que permite analizar simultáneamente a todos los estudiantes del aula desde una sola fuente de vídeo, lo cual simplifica el procesamiento y reduce la complejidad del sistema.

Sin embargo, al estudiar con mayor profundidad la viabilidad de esta aproximación, se identificaron varias limitaciones importantes. En primer lugar, la detección precisa de rostros a larga distancia y con múltiples personas en movimiento no resultaba suficientemente fiable con los modelos disponibles. Además, el análisis de atención basado en la dirección de la cabeza y el parpadeo se complicaba mucho debido a la pérdida de precisión con la distancia y los posibles obstáculos visuales como, por ejemplo, otros alumnos o el entorno que les rodeaba.

Por este motivo, se reformuló el enfoque inicial hacia una arquitectura donde cada estudiante utilizara su propio dispositivo con cámara para capturar su imagen de forma individual. Esta nueva propuesta ofrecía una mejor calidad de imagen, detección más precisa del rostro y, sobre todo, un mayor respeto por la privacidad y el anonimato, ya que la información sensible se procesaría directamente en el cliente y únicamente se envían los datos relevantes al servidor. Esta decisión también permitía ampliar el uso del sistema en desarrollos futuros a clases virtuales o híbridas, lo cual aumenta la flexibilidad del sistema.

Este cambio de enfoque al proyecto marcó un punto de inflexión en el desarrollo del proyecto, ya que obligó a rediseñar tanto las tecnologías que iban a utilizar como la interfaz de usuario, el modelo de comunicación en tiempo real, y el sistema de almacenamiento de resultados. Gracias a esta reformulación, se consiguió construir un sistema más escalable, preciso, y adaptado a distintos contextos educativos.

4.4.2. PROBLEMÁTICAS ENFRENTADAS

Este proyecto afronta unos campos que debieron ser estudiados y diseñados con cautela para conseguir un desempeño eficiente y óptimo. En este apartado, se pretende presentar los problemas principales que se han encontrado durante el desarrollo y de qué manera se ha trabajado en ellos para solucionarlos.

4.4.2.1. BALANCE DE CARGAS

Uno de los puntos más importantes del sistema era decidir cómo distribuir la carga computacional entre cliente y servidor. En un principio, se consideró que todo el procesamiento relacionado con la atención y la somnolencia se hiciera desde el servidor, centralizando así la lógica del análisis. Esto permitía que los clientes que accedían al servicio se liberaran de prácticamente la totalidad del coste computacional que existía en el análisis de la atención. Sin embargo, tras realizar pruebas con esta arquitectura, se comprobó que el sistema sufría caídas de rendimiento grandes al aumentar el número de usuarios simultáneos, además de poner en riesgo la escalabilidad futura del proyecto.

Para solucionarlo, se decidió trasladar el procesamiento con mayor carga, que era el análisis, al cliente. De esta forma, cada dispositivo ejecutaría de manera local los modelos necesarios y solo enviaría los datos clave al servidor, es decir, se enviarán los periodos de cambio de foco, periodos de somnolencia y notificaciones. Esta decisión fue clave, ya que derivó en rediseñar todo el sistema y sirvió para resolver de manera óptima y eficiente muchas de las problemáticas que surgieron más adelante.

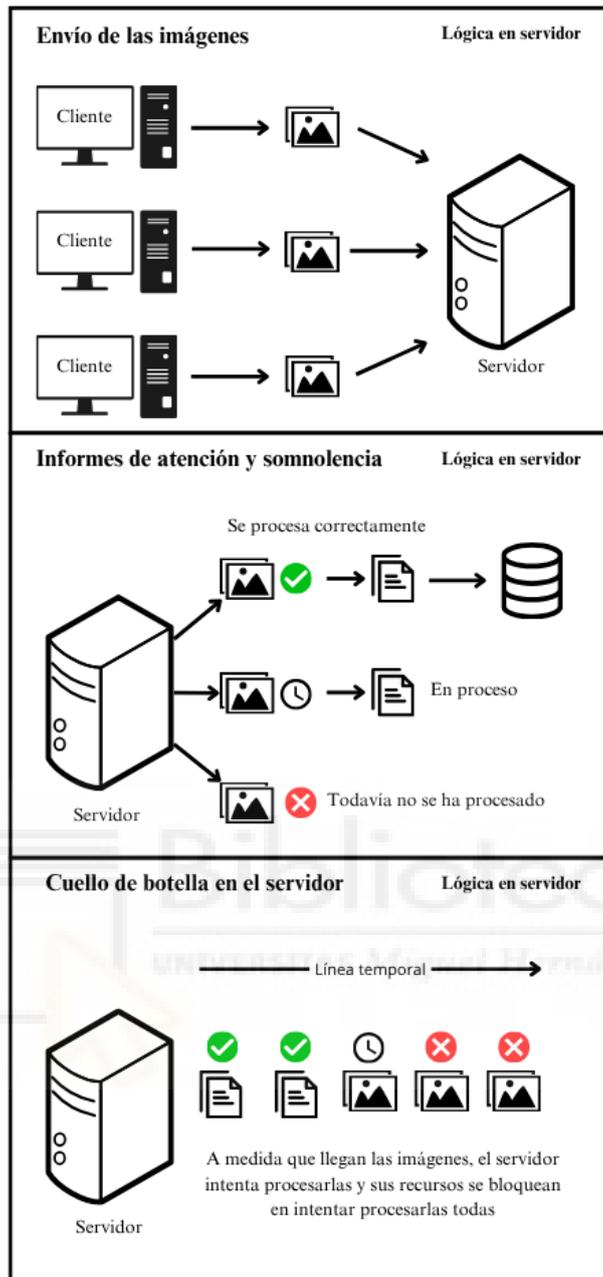


Figura 4.4—2 Problemática de una lógica centrada en el servidor

En esta alternativa se temía que ciertos dispositivos no pudieran ejecutar el modelo de inteligencia artificial, pero más tarde sorprendió dando resultados de que no solo alivió la carga del *backend*, sino que también mejoró la velocidad de respuesta, redujo la latencia y permitió mantener el sistema funcional incluso en contextos con muchos alumnos conectados a la vez.

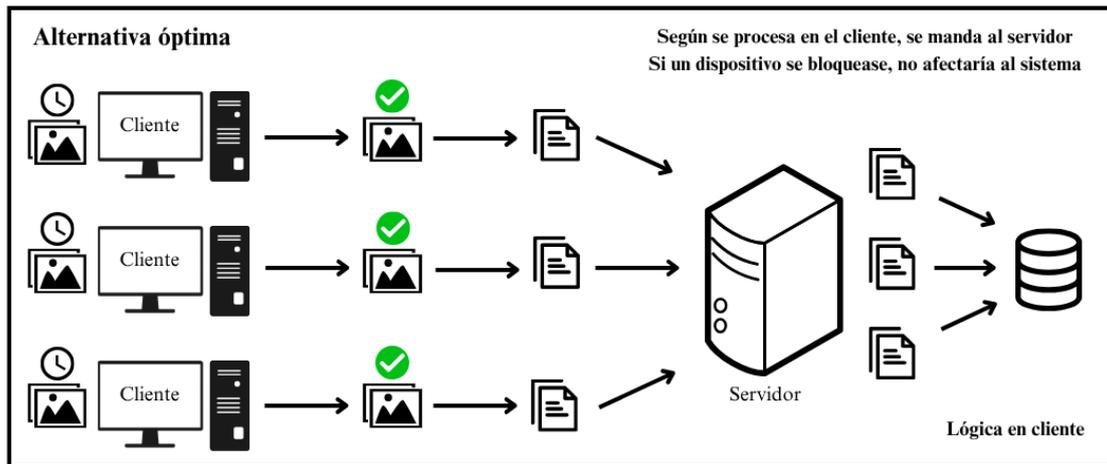


Figura 4.4—3 Optimización con la lógica centrada en el cliente

Mientras que en la alternativa anterior se debían analizar menos imágenes por segundo para compensar el coste computacional que tenía que todos los clientes mandasen sus imágenes para que fuesen analizadas, en esta alternativa se permite incluso analizar más imágenes, ya que solamente se enviará un resumen de la atención al servidor y se ha podido comprobar que el coste computacional de un solo dispositivo tiene un muy buen rendimiento y puede funcionar sin ningún tipo de problema en la gran mayoría de dispositivos, incluso los teléfonos móviles. De todas formas, la ventaja de esta arquitectura es que si, por algún motivo, un dispositivo se bloquease no afectaría al resto de componentes del sistema.

4.4.2.2. DATOS ANONIMIZADOS

Uno de los requisitos fundamentales del sistema desde su concepción fue el de preservar el anonimato de los estudiantes en todo momento. Este aspecto no solo era por motivaciones éticas, sino también por consideraciones legales relacionadas con la protección de datos personales, especialmente sensibles al tratarse de un entorno educativo. El sistema debía garantizar que ningún dato recogido pudiera ser utilizado para identificar directamente a un estudiante.

Como inicialmente se planteó que se iba a ejecutar el sistema en el servidor, se planteó utilizar las imágenes necesarias para el análisis de manera temporal, eliminándolas una vez procesadas. Sin embargo, esta solución presentaba varios inconvenientes: por un lado, presentaba un riesgo de ser expuestas en caso de brechas de seguridad; y, por otro lado, seguía suponiendo una recogida de información biométrica innecesaria. Por tanto, se decidió rediseñar el sistema para que ninguna imagen o vídeo fuera enviada ni almacenada en el servidor en ningún momento.

La decisión mencionada anteriormente de trasladar todo al lado del cliente ofrecía una ventaja importante en cuanto a privacidad: al no enviar imágenes ni secuencias de vídeo al servidor, se evitaba la manipulación de datos sensibles, alineándose mejor con las recomendaciones éticas y legales del uso de visión por computador en contextos educativos.

4.4.2.3. SELECCIÓN DE LAS MÉTRICAS

El estudio de la atención, como se ha podido ir tratando a lo largo de la memoria, es un campo muy diverso y complejo de tratar. Son muchos los factores que participan en el estado mental de estar atento, y algunas de estas métricas que miden el grado de atención son diferentes para cada persona. La atención es un estado influido por factores tanto internos, como la fatiga o las emociones; como externos, como el ruido o a cómo de interesante es aquello a lo que se presta atención.

En este proyecto se exploraron distintas alternativas a la hora de definir las métricas clave que se iban a utilizar para cuantificar la atención. Algunas de las primeras ideas incluían parámetros como el seguimiento ocular, las expresiones faciales, datos biométricos como las pulsaciones del corazón y la tensión sanguínea o incluso la postura corporal. No obstante, muchas de estas métricas presentaban problemas en su implementación práctica: o bien requerían un hardware específico, o bien no ofrecían suficiente robustez en contextos reales, como dispositivos móviles con cámaras de baja calidad, los diferentes ángulos en los que se puede poner una cámara o la luz del entorno.

Por todo ello, que se optó por una estrategia mucho más flexible y centrada en un uso más realista, en el que no se necesitan grandes recursos económicos para funcionar y en cualquier lugar, con cualquier dispositivo, pudiese hacer uso de una herramienta que, si bien no proporciona una fiabilidad perfecta, se acerca y en gran medida a la verdad. Gracias a que la atención viene, en gran medida, a la dirección de la cabeza y gracias a que la somnolencia es significativamente notable con el parpadeo y la apertura ocular, se lograron conseguir resultados excelentes y útiles.

4.4.2.4. GESTIÓN DE MÚLTIPLES CONEXIONES SIMULTÁNEAS

La aplicación permite que múltiples estudiantes se unan simultáneamente a una clase. Cada uno de ellos transmite datos en tiempo real, como ya se ha podido ver anteriormente, lo cual genera un alto volumen de conexiones activas al servidor. Para gestionar esta información, se ha implementado una arquitectura basada en Sockets que permite una comunicación bidireccional eficiente.

Cada vez que un estudiante entra en una clase, se establece una conexión mediante WebSockets. Esta conexión permanece abierta durante toda la sesión y transmite los datos relevantes de atención de forma periódica. El servidor identifica a cada usuario, sea profesor o alumno, mediante un identificador único compuesto por su alias y la sala en la que se encuentra. Este diseño permite llevar control de quién hace qué evento en cada momento, así como también permite el envío y recepción de datos entre cliente y servidor.

Para evitar cuellos de botella, los datos se almacenan cada minuto en el servidor. Sin embargo, el envío de las notificaciones ha sido optimizado para poder funcionar en tiempo real, por lo que, si un alumno desencadena una señal para que se envíe una notificación de indicios de sueño o falta de atención, esta será enviada inmediatamente al profesor.

Se ha probado el sistema localmente con un número reducido de usuarios simultáneos (entre 5 y 10) sin pérdida de rendimiento. Sin embargo, la estructura está pensada para poder escalar horizontalmente, es decir, añadiendo mejores máquinas o incluso desplegando en contenedores que optimicen los recursos.

El sistema, además, está preparado para las caídas, ya que cuenta con mecanismos de reconexión automáticos. También está preparado para enviar de manera asíncrona los datos y así evitar bloqueos o recargas de la página. Es por todo ello por lo que el usuario no se deberá de preocupar, puede dejar el servicio web ejecutándose sin necesidad de preocuparse en volver a conectarse.

4.5. RESULTADOS FINALES

En este apartado se detallarán cuáles han sido los resultados obtenidos de los apartados principales de la página web. No se hará hincapié en aquellas secciones del servicio que sean puramente estéticas o que no sean relevantes por hallarse fuera del objetivo principal del proyecto. Las características que tiene el software y que no serán mostradas por no ser relevantes incluyen las páginas auxiliares o decorativas del sitio web, las animaciones o estilos dinámicos de los elementos del sitio, o también la adaptación a todo tipo de resoluciones que tienen todas y cada una de las páginas

4.5.1. PÁGINA PRINCIPAL

En la Figura 4.5—1 Página principal del sitio web se puede observar que el estilo del sitio es muy minimalista, e intenta buscar que los elementos saturen lo mínimo posible al usuario. El diseño general es moderno y minimalista, priorizando la usabilidad.

En cuanto a su estructura, la página cuenta con una sección de inicio de sesión por la cual el usuario puede elegir su rol y, si es estudiante, un alias y la clase a la cual se quiere unir, seguido de un botón de inicio de sesión.

Más abajo en la página tiene elementos cosméticos, pues es un apartado en el que se habla del proyecto y se hace una demostración de cómo funciona el servicio web y qué datos recopila.


Focus AI
ASISTENTE EDUCATIVO

 Consejos de atención

¡Te damos la bienvenida!

Escoge un rol y piensa un alias.



Profesor



Estudiante

Entrar a la clase

Sobre nosotros



¿Qué es FocusAI?

Es una herramienta en línea que permite a los profesores analizar cuánta atención prestan sus estudiantes en tiempo real, utilizando tecnología de inteligencia artificial.

¿Por qué FocusAI?

Sus fines son investigativos y educativos, permitiendo a los profesores mejorar la calidad de sus clases mediante datos objetivos y análisis en tiempo real.

Potencia la Concentración

FocusAI te ayuda a aprender sobre la atención de tus estudiantes



Medición del Foco

Análisis preciso de los niveles de concentración durante la clase



Patrones de Atención

Identifica momentos clave de engagement y desconexión



Mejora Continua

Optimiza tus clases basándote en datos de atención reales

Habilitar webcam

● Monitoreo en Tiempo Real

¿Cómo lo analizamos?

- Somos transparentes. Solamente recopilamos los datos más fundamentales para el análisis de la atención
- Recibe alertas inteligentes cuando los niveles de concentración disminuyen




2025 Focus AI - Javier Maciá Sempere

Construido con  en Alicante, España
Protección de datos 
Política de cookies 

Figura 4.5—1 Página principal del sitio web

4.5.2. PÁGINA DE REGISTRO

Si se selecciona el botón de “Profesor” en la página de inicio de sesión, será posible registrarse o iniciar sesión en el acceso identificado del sitio web. Desde aquí, se da la posibilidad de registrarse si el usuario no lo ha hecho.

Figura 4.5—2 Acceso como profesor

Desde el registro, los datos que se piden para iniciar una cuenta consisten en un nombre de usuario y una contraseña. Al pulsar en “registrarse” se accede directamente a la página de inicio de sesión.

Figura 4.5—3 Página de registro

4.5.3. PANEL DEL PROFESOR

Como se ha podido observar, también es posible iniciar sesión como profesor. Si se tiene una cuenta creada y se ingresa el nombre de usuario y la contraseña, se podrá visualizar el panel del profesor.

En la Figura 4.5—4 Página principal del acceso identificado, el profesor tiene la posibilidad de crear una clase poniendo únicamente los campos de nombre y de descripción. Desde el panel principal, se puede cerrar la sesión o acceder a la ventana de estadísticas y perfil.

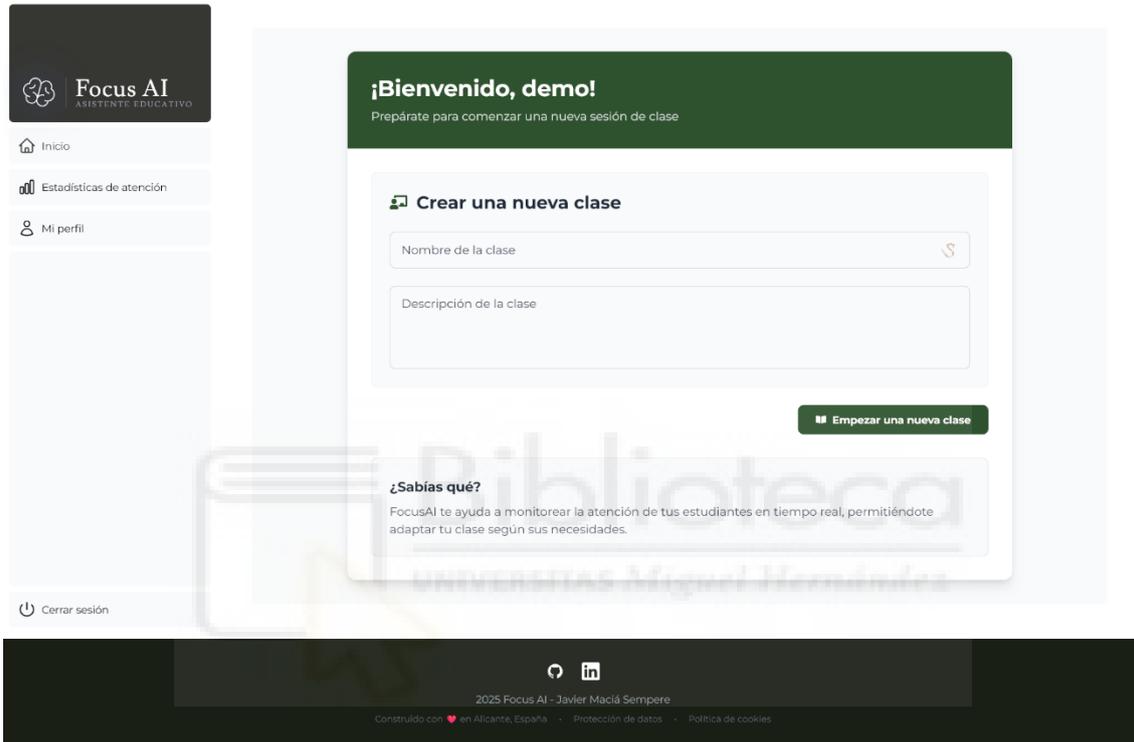


Figura 4.5—4 Página principal del acceso identificado

En la página de estadísticas de la atención, si se ha creado una clase con alumnos, será posible ver un análisis de la atención de manera global e individual, con estadísticas temporales de cómo ha funcionado la clase a lo largo del tiempo y de en qué momentos los alumnos han desviado su atención o han mostrado signos de estar entrando en el sueño.

En la Figura 4.5—5 Resultados de la atención de una clase es posible observar un elemento para seleccionar una clase, y a continuación el resto de las estadísticas, en orden:

1. **Resumen de la sesión:** con datos principales sobre la clase y su atención media.
2. **Niveles de atención por alumno:** con un desglose por alumno de su porcentaje de atención, cambio de foco y signos de sueño.
3. **Eventos totales por momento:** representa cuántos eventos de somnolencia y cambio de atención ha habido durante la clase.
4. **Distribución horaria de eventos:** igual que el anterior, pero agrupado por franjas de tiempo en gráficos de barras.



- [Inicio](#)
- [Estadísticas de atención](#)
- [Mi perfil](#)

[Cerrar sesión](#)

Estadísticas de Sesiones

Seleccionar Sesión

Clase de demostración - 4 de mayo de 2025, 21:28

Resumen de la Sesión

Duración de la sesión: 0 horas, 19 minutos y 50 segundos

88.7%
Atención Media

2
Estudiantes Analizados

12
Cambios de foco

17
Signos de sueño

Duración media de desfocalización

3.2 segundos

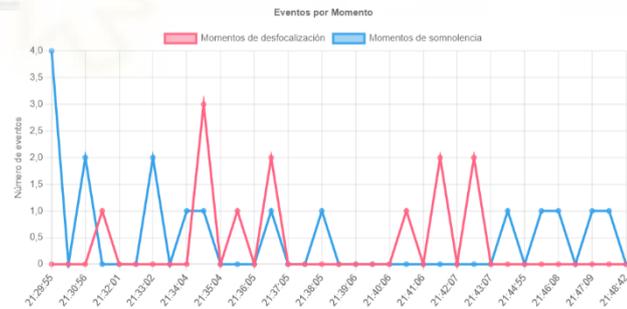
Duración media de somnolencia

11.7 segundos

Niveles de atención por alumno



Eventos totales por momento



Distribución horaria de eventos

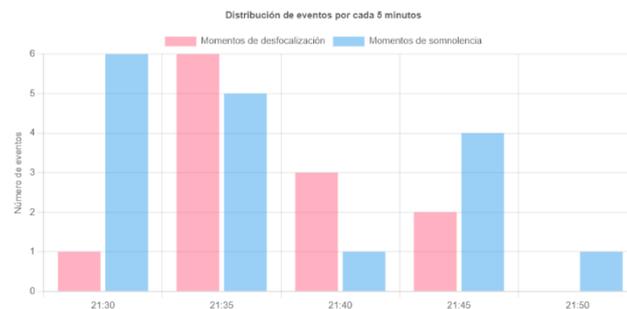


Figura 4.5—5 Resultados de la atención de una clase

Si el profesor accede al apartado de “perfil”, podrá cambiar su alias, contraseña o eliminar su cuenta.

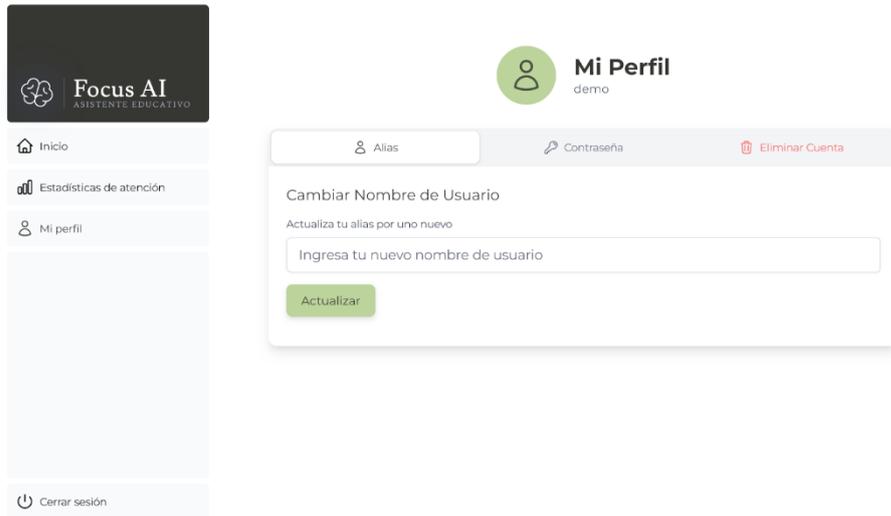


Figura 4.5—6 Cambiar alias del profesor



Figura 4.5—7 Cambiar contraseña del profesor

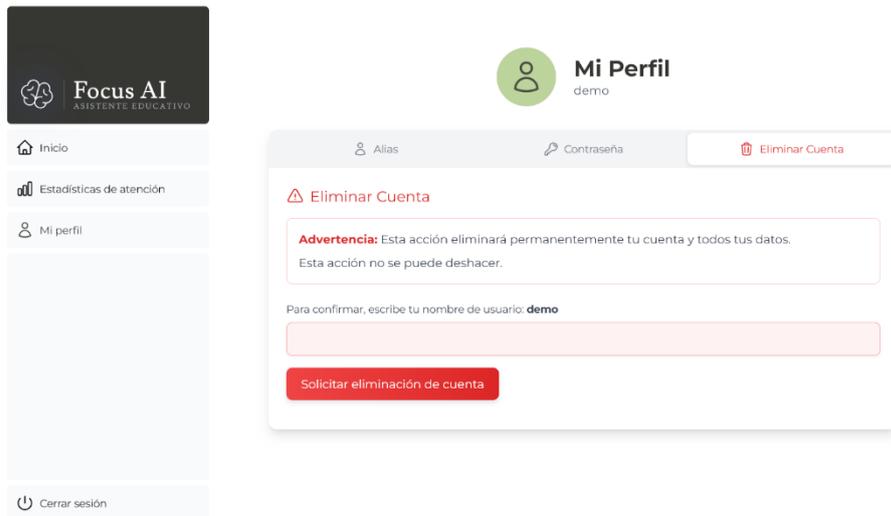


Figura 4.5—8 Eliminar cuenta del profesor

4.5.4. CLASE

La interfaz de la página Clase es diferente según el rol que tome el usuario. Si se opta por ser profesor, se mostrará una ventana por la cual el profesor podrá ver tanto los detalles de la sesión como las notificaciones recibidas, el tiempo de clase o los participantes que hay en ella.

Desde el panel de alertas, el profesor podrá saber quién y a qué hora ha cambiado su foco de atención o ha mostrado signos de somnolencia. También podrá activar o desactivar el sonido de notificación. Tendrá habilitada también la opción de salir de la clase, acción la cual provocará que esta acabe.

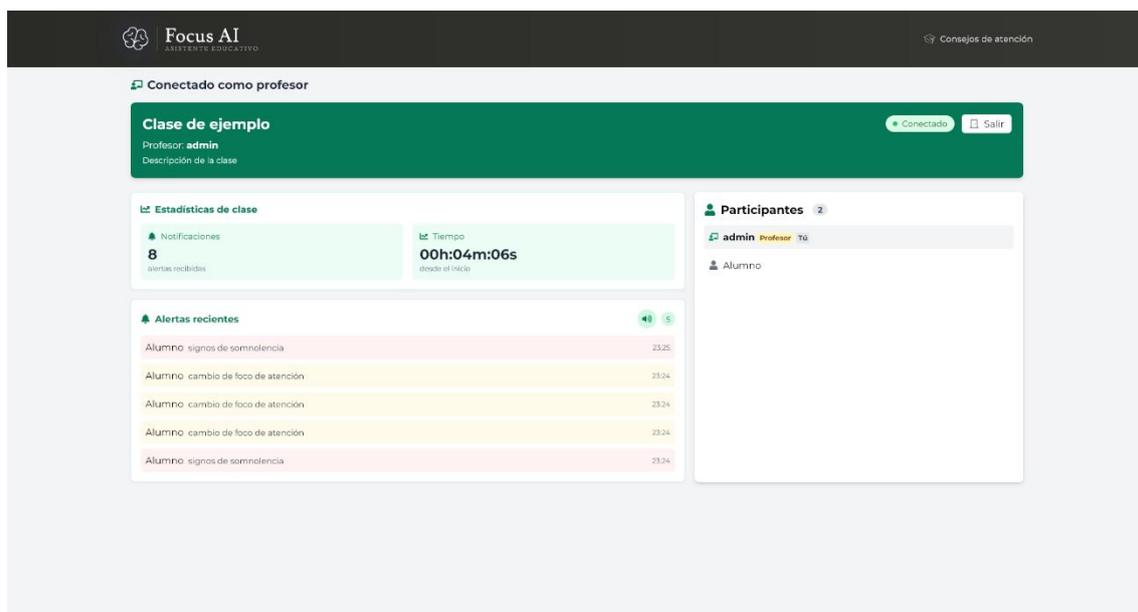


Figura 4.5—9 Interfaz de la clase del profesor

Si a la clase accede un usuario con el rol de alumno, su interfaz se verá diferente. Para este tipo de usuario, únicamente se mostrarán los detalles de nombre, creador y descripción de la clase, y podrá ver su cámara junto con la lista de los participantes que hay en la clase. También tendrá una opción para salir de la clase habilitada.

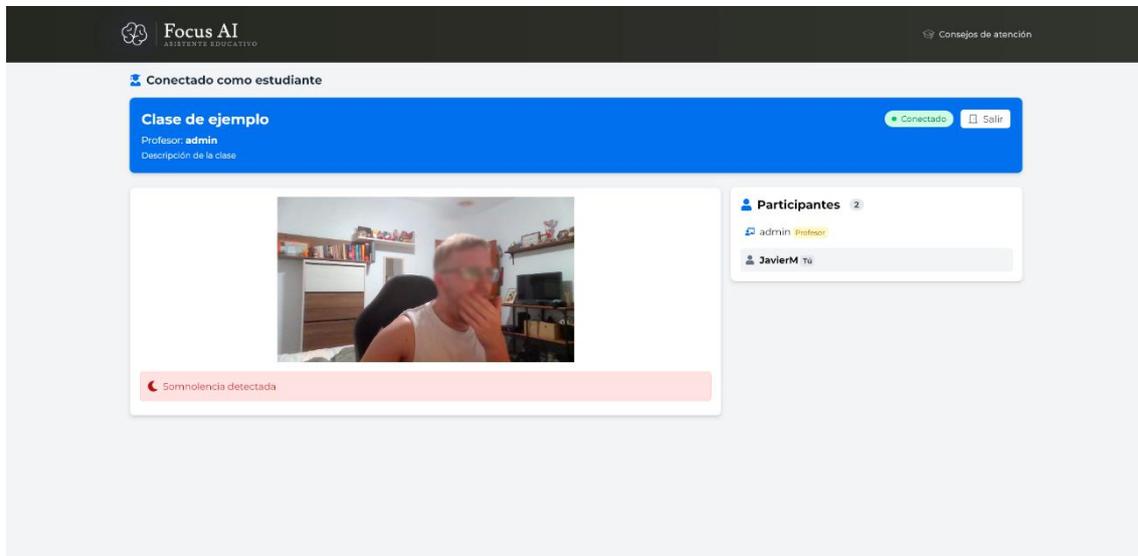


Figura 4.5—10 Interfaz del alumno ante signos de somnolencia

En ambas figuras, Figura 4.5—10 Interfaz del alumno ante signos de somnolencia y Figura 4.5—11 Interfaz del usuario ante cambios en la atención es posible observar que al alumnado también se le notificará cuando se detecte un evento de uno de los dos tipos disponibles.

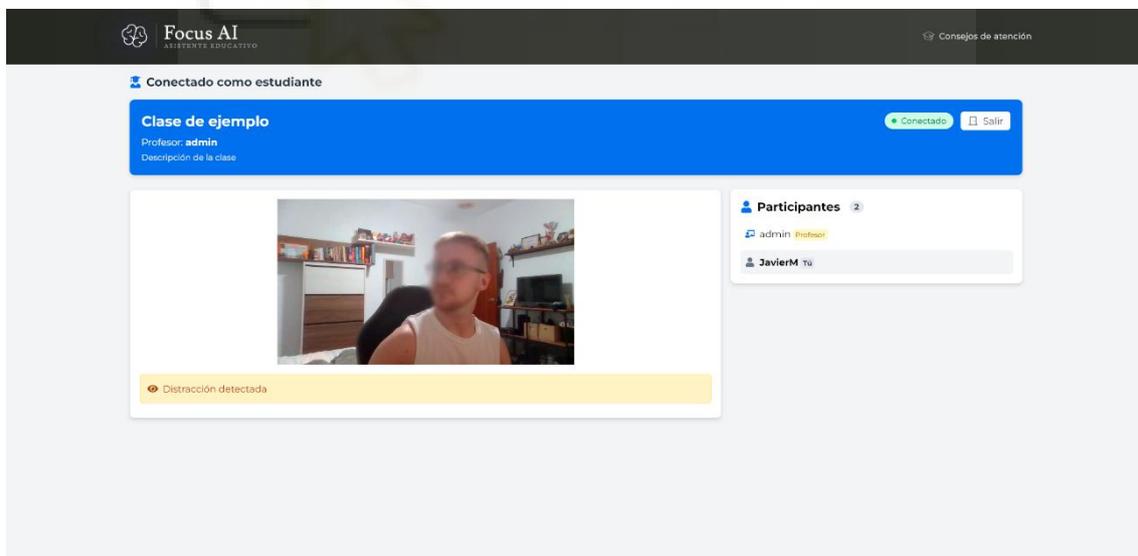


Figura 4.5—11 Interfaz del usuario ante cambios en la atención

Capítulo 5

Conclusiones



Biblioteca
UNIVERSITATIS M...

5. CONCLUSIONES

En este Trabajo de Fin de Grado se ha podido abordar un desafío en el cual se debía monitorizar la atención del usuario en un entorno digital, desarrollando una aplicación que aprovecha las capacidades de la visión por computador y la inteligencia artificial. A través de la implementación de algoritmos de procesamiento de imagen, se ha logrado construir un sistema capaz de analizar en tiempo real métricas clave de la atención, como la dirección de la mirada, la posición de la cabeza y la frecuencia de parpadeo, proporcionando una estimación robusta del nivel de concentración del usuario.

Se han cumplido con los objetivos de crear una aplicación web que pudiese medir la atención de las personas, en tiempo real y por medio de una interfaz web intuitiva y fácil de usar. Además, permite respetar la privacidad de los usuarios por medio de pseudónimos y es de utilidad para el profesor tanto durante las clases como después, ya que envía notificaciones en tiempo real pero también genera resúmenes a los alumnos. No se ha cumplido el objetivo de analizar las emociones únicamente por no resultar tan relevante para los costes computacionales que conlleva y por falta de tiempo en la entrega del proyecto.

Los resultados obtenidos validan que existe una viabilidad técnica de la propuesta que, además, está construida de modo que presenta un potencial de ser una herramienta que pueda ser mejorada para abarcar más contextos de caso de uso. El potencial de la aplicación no solo demuestra que los diferentes conceptos trabajados se puedan limitar a los entornos educativos, donde la monitorización de la atención podría optimizar el aprendizaje y detectar la fatiga temprana, sino que también se podría extender al ámbito de la seguridad, por ejemplo, en la detección de somnolencia al volante.

5.1. AMPLIACIONES FUTURAS

Una de las ventajas principales con las que cuenta el proyecto es su gran potencial de ser ampliado. Gracias a su modularidad y a su contexto de aplicación web orientada a la educación y a la investigación de la atención, existen muchos campos en los que se puede extender la funcionalidad de la aplicación a más ámbitos del mundo educativo y científico. Las alternativas presentadas en este apartado, si bien son interesantes, no han sido profundizadas porque existían limitaciones de tiempo para presentar el proyecto y/o además quedaban fuera del ámbito del caso de uso que se perseguía. Algunas propuestas presentaban, además, unas limitaciones económicas importantes.

Puesto que el objetivo principal del proyecto es servir de herramienta auxiliar a los profesores o entidades educativas para analizar la atención de sus alumnos, una propuesta importante de este proyecto sería contar con un sistema de gestión de usuarios y permisos mucho más avanzado. La idea principal de esta propuesta sería la de crear una entidad por encima de los profesores que tenga permisos para poder borrar profesores y datos relevantes a sus cuentas. Además, el sistema, a modo de proteger que cualquiera se pueda crear una cuenta como profesor, podría verificar que únicamente aquellos que sean docentes puedan crear cuentas de usuario, autenticándose con la cuenta de correo educativa o similares.

Una ampliación interesante que se barajó en un inicio fue la de contemplar otros indicativos físicos y medibles de pérdida de atención y/o somnolencia. Se contempló la posibilidad de medir las emociones que sentía cada usuario, o hacer uso de dispositivos como relojes inteligentes que fuesen capaces de medir datos biométricos como la presión en la sangre o las pulsaciones. Esta propuesta requiere de una investigación exhaustiva para conocer si realmente influyen tanto estos indicativos de la atención como para incluirlos en el programa, así como también el análisis de si un sistema doméstico podría funcionar en óptimas condiciones con estos procesos añadidos.

Finalmente, otra de las ampliaciones importantes de este software podría ser, también, la de implementar interfaces por las cuales la aplicación se pueda conectar e integrar con otras aplicaciones (APIs). Esto podría ser, por ejemplo, teniendo una manera de poder enviar los resúmenes de las sesiones online a otra plataforma que permita almacenarlas o estudiarlas.

Como se puede ver, incluso con los objetivos cumplidos, este software puede ser ampliado de muchas maneras diferentes en un futuro para abarcar más campos nuevos o mejorar los ya existentes.





Anexos

ANEXO I: MANUAL DE INSTALACIÓN

Para realizar la instalación del software, hay varios requisitos que se deben satisfacer previamente. Sin estos requisitos, el sistema no podrá realizar sus funciones básicas. Lo que se necesitará obtener es:

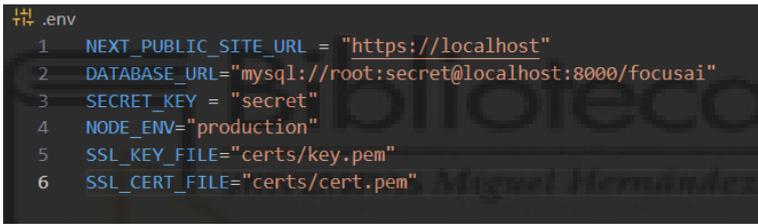
- **Código fuente** del sistema.
- Entorno de ejecución de **Node.js** en su versión 20 o superior, obtenible desde su página oficial (<https://nodejs.org/es/download>).
- **MySQL**, en su versión 8.0 o superior. Obtenible desde su página oficial (<https://dev.mysql.com/downloads/workbench/>).

Desde la carpeta raíz donde se tenga alojado el código fuente, se deberá ejecutar el siguiente comando por la línea de comandos:

```
> npm install
```

Esto instalará todas las dependencias que son necesarias para que el proyecto funcione.

A continuación, se deberán configurar las variables de entorno. Para ello, en el archivo raíz, existe un archivo llamado “.env.example”. Este archivo se deberá copiar y renombrar a “.env” en el mismo directorio. El fichero tendrá este aspecto:



```

+ .env
1  NEXT_PUBLIC_SITE_URL = "https://localhost"
2  DATABASE_URL="mysql://root:secret@localhost:8000/focusai"
3  SECRET_KEY = "secret"
4  NODE_ENV="production"
5  SSL_KEY_FILE="certs/key.pem"
6  SSL_CERT_FILE="certs/cert.pem"
  
```

Figura 0—1 Contenido del archivo .env

- **NEXT_PUBLIC_SITE_URL:** define la dirección del sitio. Si se quiere ejecutar el software para que el resto de usuarios de una red puedan acceder a él, se podrá poner la dirección IP de la máquina. Para saber la IP de la máquina desde Windows, se ejecutará el comando *ipconfig*.
- **DATABASE_URL:** es la URL de conexión a una base de datos. Está compuesto por *[Protocolo]://[Usuario de base de datos]:[Contraseña del usuario]@[Dirección del servidor de la base de datos]:[Puerto]/[Nombre de la base de datos]*
- **SECRET_KEY:** es una clave que hace la función de firma y que se usa a modo de verificación para comprobar la autenticación de los usuarios.
- **NODE_ENV:** indica si el proyecto se está ejecutando en producción (*production*) o desarrollo (*development*). Útil en ciertas tareas internas.
- **SSL_KEY_FILE:** es la localización donde se encuentra la clave para el cifrado de la conexión.
- **SSL_CERT_FILE:** es la localización donde se encuentra el certificado para el cifrado de la conexión.

Lo siguiente que se debe hacer es configurar la base de datos. Para ello, únicamente hay que ejecutar los comandos:

> *npx prisma generate* # para generar el cliente de Prisma.

> *npx prisma db push* # para crear la base de datos.

```

C:\Users\Usuario\Desktop\FocusAI>npx prisma generate
Environment variables loaded from .env
Prisma schema loaded from prisma\schema.prisma

✔Generated Prisma Client (v5.19.1) to .\node_modules\@prisma\client in 108ms
Start by importing your Prisma Client (See: http://pris.ly/d/importing-client)
Tip: Interested in query caching in just a few lines of code? Try Accelerate today! https://pris.ly/tip-3-accelerate

Update available 5.19.1 -> 6.8.2
This is a major update - please follow the guide at
https://pris.ly/d/major-version-upgrade

Run the following to update
  npm i --save-dev prisma@latest
  npm i @prisma/client@latest

C:\Users\Usuario\Desktop\FocusAI>npx prisma db push
Environment variables loaded from .env
Prisma schema loaded from prisma\schema.prisma
Datasource "db": MySQL database "focusai" at "localhost:8000"

The database is already in sync with the Prisma schema.

✔Generated Prisma Client (v5.19.1) to .\node_modules\@prisma\client in 83ms

C:\Users\Usuario\Desktop\FocusAI>

```

Figura 0—2 Crear la base de datos

El siguiente paso es agregar el certificado y la clave a la carpeta previamente indicada. No es recomendable, pero es posible autofirmar estos dos archivos para hacer funcionar el software. Para ello, se pueden utilizar programas como OpenSSL o la herramienta de Microsoft *makecert*.

A continuación, habrá que compilar el proyecto. Para realizar este paso, solamente se tendrá que ejecutar el siguiente comando:

> *npm run build*

Finalmente, para que pueda funcionar el servicio web, se deberá ejecutar:

> *node server.js*

Con todo esto, el servicio quedará en marcha y cualquier persona de la red podrá acceder a él.

Bibliografía



6. BIBLIOGRAFÍA

- [1] P. M. H. D. Vasto, «Influencia de las tecnologías de información y comunicación (TIC) en el proceso enseñanza-aprendizaje: una mejora de las competencias digitales,» *Revista Científica General José María Córdova*, vol. 13, nº 16, pp. 121-132, 2015.
- [2] Gobierno de España, «Qué es la Inteligencia Artificial,» [En línea]. Available: <https://planderecuperacion.gob.es/noticias/que-es-inteligencia-artificial-ia-prtr>. [Último acceso: Abril 2024].
- [3] A. M. Turing, «I.—COMPUTING MACHINERY AND INTELLIGENCE,» *Mind*, vol. LIX, nº 236, pp. 433-460, 190.
- [4] Microsoft, «¿Qué es el aprendizaje automático?,» [En línea]. Available: <https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-machine-learning-platform>. [Último acceso: Abril 2024].
- [5] IBM, «¿Qué es son las redes neuronales?,» [En línea]. Available: <https://www.ibm.com/es-es/topics/neural-networks>. [Último acceso: Abril 2024].
- [6] IBM, «¿Qué es Deep Learning?,» [En línea]. Available: <https://www.ibm.com/es-es/topics/deep-learning>. [Último acceso: Abril 2024].
- [7] Supervisor Europeo de Protección de Datos, «Facial Emotion Recognition,» [En línea]. Available: https://www.edps.europa.eu/system/files/2021-05/21-05-26_techdispatch-facial-emotion-recognition_ref_en.pdf. [Último acceso: Marzo 2024].
- [8] Real Academia Española, «Diccionario esencial de la lengua española, Atender,» [En línea]. Available: <https://www.rae.es/desen/atender>. [Último acceso: Febrero 2024].
- [9] A. Estévez-González, C. García-Sánchez y C. Junqué, «La atención: una compleja función cerebral,» *Revista de Neurología*, vol. 25, nº 148, pp. 1989-1997, 1997.
- [10] L. Marquez Carpintero, M. Pina Navarro, S. Suescun Ferrandiz, F. Escalona, F. Gomez Donoso, R. Roig Vila y M. Cazorla, «Artificial Intelligence-based System for Detecting Attention Levels in Students,» *J. Vis. Exp.*, 2023.
- [11] D. Cazzato, M. Leo, C. Distante y H. Voos, «When I Look into Your Eyes: A Survey on Computer Vision Contributions for Human Gaze Estimation and Tracking,» *Sensors*, 2020.
- [12] M. Raca, Ł. Kidziński y P. Dillenbourg, «Translating head motion into attention-towards processing of student's body-language,» de *Proceedings of the 8th International Conference on Educational Data Mining*, 2015.

- [13] J. In Koh, S. Ray, J. Cherian, P. Taelle y T. Hammond, «Gauging attention of students in an e-learning environment,» de *2020 IEEE 4th Conference on Information & Communication Technology (CICT)*, Chennai, India, 2020.
- [14] S. Clemente, «La mente es maravillosa,» 16 Noviembre 2017. [En línea]. Available: <https://lamenteesmaravillosa.com/las-microexpresiones-segun-paul-ekman/>. [Último acceso: Marzo 2024].
- [15] G. Mark, D. Gudith y U. Klocke, «The cost of interrupted work: More speed and stress,» de *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Florence, Italy, 2008.
- [16] R. Mayor, E. y J. Rey De Castro Mujica, «Somnolencia: Qué es, qué la causa y cómo se mide,» *Acta médica peruana*, vol. 27, pp. 137-143, 2010.
- [17] M. Nyström, R. Andersson, D. Niehorster, R. Hessels y I. Hooge, «What is a blink? Classifying and characterizing blinks in eye openness signals,» *Behav Res Methods*, 2024.
- [18] PrepLadder, «Blinking Of Eye: Types, Uses, Causes, Diagnosis and Treatment,» 13 Marzo 2024. [En línea]. Available: <https://www.prepladder.com/neet-pg-study-material/ophthalmology/blinking-of-eye-types-uses-causes-diagnosis-and-treatment>. [Último acceso: Mayo 2025].
- [19] Mirage News, «Mirage News,» 6 Junio 2023. [En línea]. Available: <https://www.miragenews.com/behind-the-blink-the-science-of-eye-blinking-1021184/>. [Último acceso: Mayo 2025].
- [20] T. Nakano, M. Kato, Y. Morito, S. Itoi y S. Kitazawa, «Blink-related momentary activation of the default mode network while viewing videos,» *Proc Natl Acad Sci U S A.*, 2013.
- [21] G. Barbato, G. Ficca, G. Muscettola y M. T. Fichelle, «Diurnal variation in spontaneous eye-blink rate,» *Psychiatry Research*, 2000.
- [22] S. Pérez García, «Detección de somnolencia durante la conducción en un ambiente simulado.,» 2022.
- [23] A. Bahill, M. Clark y L. Stark, «Computer simulation of overshoot in saccadic eye movements *Comput Programs Biomed*,» 1975.
- [24] M. Rodríguez Montiel, «Estudio de las características del parpadeo, y su relación con los movimientos sacádicos, en distintas condiciones controladas de lectura,» 2015.
- [25] mdn web docs, «HTML: Lenguaje de etiquetas de hipertexto,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/HTML>. [Último acceso: Julio 2024].

- [26] mdn web docs, «CSS,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/CSS>. [Último acceso: Julio 2024].
- [27] TailwindCSS, «Sitio oficial de Tailwindcss,» [En línea]. Available: <https://tailwindcss.com/>. [Último acceso: Julio 2024].
- [28] mdn web docs, «JavaScript,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/JavaScript> .
- [29] TypeScript, «Sitio oficial de TypeScript,» [En línea]. Available: <https://www.typescriptlang.org/>. [Último acceso: Julio 2024].
- [30] React, «Sitio oficial de React,» [En línea]. Available: <https://es.react.dev/>. [Último acceso: Julio 2024].
- [31] NextJS, «Sitio oficial de Next.js,» [En línea]. Available: <https://nextjs.org/>. [Último acceso: Julio 2024].
- [32] NodeJS, «Sitio oficial de NodeJS,» [En línea]. Available: <https://nodejs.org/en>. [Último acceso: Julio 2024].
- [33] MySQL, «Sitio oficial de MySQL,» [En línea]. Available: <https://www.mysql.com/>. [Último acceso: Julio 2024].
- [34] Datademia, «¿Qué es SQL?,» [En línea]. Available: <https://datademia.es/blog/que-es-sql>. [Último acceso: Julio 2024].
- [35] Socket.IO, «Sitio oficial de Socket.IO,» [En línea]. Available: <https://socket.io/>. [Último acceso: Julio 2025].
- [36] Prisma, «Sitio oficial de Prisma,» [En línea]. Available: <https://www.prisma.io/>. [Último acceso: Julio 2024].
- [37] Google, «Guía de soluciones de MediaPipe,» [En línea]. Available: <https://ai.google.dev/edge/mediapipe/solutions/guide?hl=es-419>. [Último acceso: Mayo 2024].
- [38] Trello, «Trello,» [En línea]. Available: <https://trello.com/>. [Último acceso: Junio 2025].
- [39] E. G. Maida y J. Pacienza, «Metodologías de desarrollo de software,» 2015. [En línea]. Available: <https://repositorio.uca.edu.ar/handle/123456789/522>. [Último acceso: Mayo 2025].
- [40] L. Bennett, «Guru99,» 13 Agosto 2024. [En línea]. Available: <https://www.guru99.com/es/software-engineering-prototyping-model.html>. [Último acceso: 2025].
- [41] V. Solutions, «Definición de requisitos: ¿Qué es y cómo aplicarlo? | Guía completa,» [En línea]. Available:

- <https://visuresolutions.com/es/blog/requirements-definition/>. [Último acceso: Mayo 2025].
- [42] IBM, «Página oficial de IBM,» [En línea]. Available: <https://www.ibm.com/docs/es/product-master/12.0.0?topic=processes-defining-use-cases>. [Último acceso: Mayo 2025].
- [43] J. Rumbaugh, *El lenguaje unificado de modelado*, Madrid: Pearson Education, S.A., 2007.
- [44] Wikipedia, «Frecuencia de muestreo,» [En línea]. Available: https://es.wikipedia.org/wiki/Frecuencia_de_muestreo. [Último acceso: Junio 2025].
- [45] Intel, «Cámara con sensor de profundidad Intel® RealSense™ D435,» [En línea]. Available: <https://www.intel.la/content/www/xl/es/products/sku/128255/intel-realsense-depth-camera-d435/specifications.html>. [Último acceso: Mayo 2025].
- [46] S. Ali I., S. Naglaa F., A. Abeer D., A. E.-S. Fathi E. y S. Ahmed, «Deploying Machine Learning Techniques for Human,» *Computational Intelligence and Neuroscience*, 2022.
- [47] P. Ekman y F. Wallace V, «Facial action coding system,» *Environmental Psychology & Nonverbal Behavior*, 1978.
- [48] J. Martínez Madruga, «Sistema de detección de emociones faciales mediante técnicas de Machine Learning adaptado a ROS para un robot de bajo coste basado en Raspberry Pi,» Universidad Rey Juan Carlos, 2022.
- [49] A. I. Siam, N. F. Soliman, A. D. Algarni, F. E. Abd El-Samie y A. Sedik, «Deploying Machine Learning Techniques for Human Emotion Detection,» *Computational Intelligence and Neuroscience*, 2022.
- [50] Amazon, «¿Qué es una red neuronal?,» [En línea]. Available: <https://aws.amazon.com/es/what-is/neural-network/>. [Último acceso: Abril 2024].
- [51] Aplyca, «NextJS: ¿el futuro de la web?,» [En línea]. Available: <https://www.aplyca.com/blog/nextjs-el-futuro-web-que-es-nextjs>. [Último acceso: Julio 2024].