

Instituto Tecnológico y de Estudios Superiores de Monterrey

Inteligencia Artificial Avanzada para la Ciencia de Datos II (Gpo 501)



**Tecnológico
de Monterrey**

Momento de Retroalimentación: Reto Datos

Equipo 5

Jorge Eduardo De León Reyna - A00829759

David Esquer Ramos - A01114940

Francisco Mestizo Hernández - A01731549

Adrián Emmanuel Faz Mercado - A01570770

Octubre 11, 2023

Servicios de nube seleccionados

Para este proyecto se seleccionó el uso de las herramientas de **Amazon Web Services (AWS)** gracias a su variedad de servicios y herramientas además de la extensa comunidad de apoyo y soporte en la cual nos podremos apoyar para el desarrollo de la herramienta. A continuación se enlistan las herramientas a utilizar en AWS:

1. **Almacenamiento - AWS Simple Storage Service (S3):** Almacenamiento de imágenes y modelos para el funcionamiento de la aplicación.
2. **Web Hosting - AWS Lightsail:** Hosting y administración de la Aplicación Web de monitoreo de la plataforma.

Por otro lado, para el almacenamiento de los datos de los estudiantes, incluyendo sus asistencias y sus participaciones, se utilizará el servicio de **MongoDB Atlas**.

MongoDB es una base de datos no relacional orientada a documentos, la cual proporciona flexibilidad y escalabilidad en el almacenamiento de información. A diferencia de las bases de datos relacionales tradicionales, MongoDB permite almacenar datos en documentos flexibles similares a JSON, lo que significa que los campos pueden variar entre los documentos y la estructura de datos puede ser modificada con el tiempo.

Se seleccionó este servicio en la nube porque al ser no relacional, proporciona una mayor flexibilidad al no tener un esquema fijo. En el caso de nuestro proyecto, los datos de los estudiantes, sus asistencias y participaciones no siempre seguirán una estructura uniforme, por lo que la flexibilidad de MongoDB facilita ese punto. Además, una de sus mayores ventajas es que ofrece una versión gratuita, permitiendo que almacenemos nuestros datos sin costos iniciales, principalmente porque no se trata de una gran cantidad de datos.

De igual forma, al estar en la nube, el servicio se ajusta automáticamente a las demandas. En caso de que un día exista un aumento en la cantidad de datos o estudiantes, el sistema se puede adaptar sin problema. MongoDB incluye copias de seguridad y cuenta con la posibilidad de acceder a la información desde cualquier lugar, facilitando el trabajo de equipo y la toma de decisiones.

Tipo de datos a utilizar

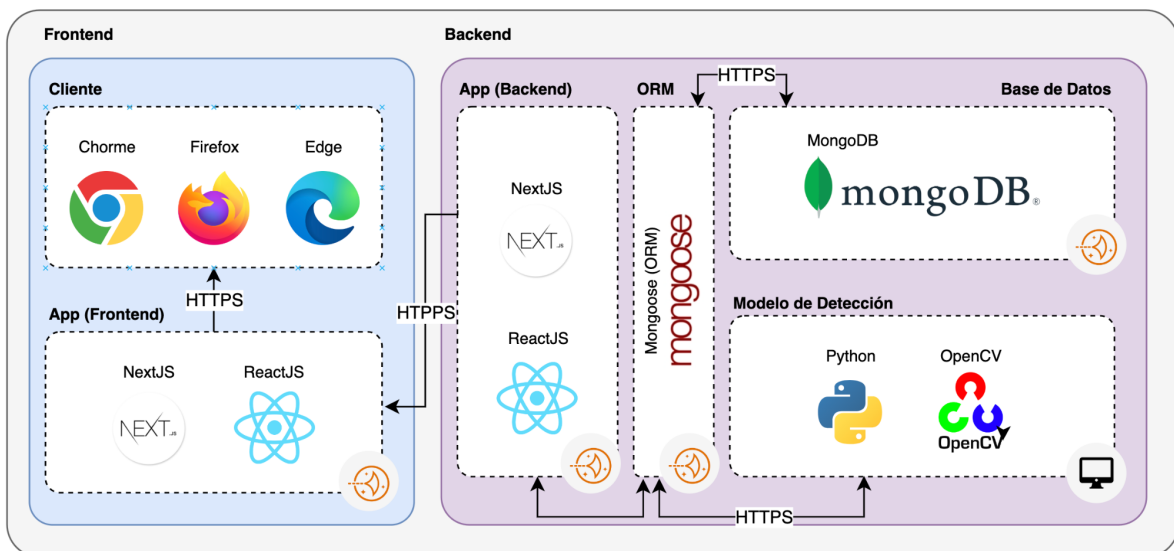
Los datos que se están almacenando en este proyecto están relacionados con la información de las asistencias y participaciones de los alumnos en los diferentes cursos. Al estar utilizando MongoDB para almacenar estos datos, se está trabajando con un almacenamiento de **block storage**, el cual consiste en el almacenamiento de datos en bloques fijos en un entorno de disco. Cada bloque actúa como un contenedor individual de datos y tiene su propia dirección, lo que permite un acceso rápido y eficiente a partes específicas de los datos, sin tener que leer todo el conjunto.

Además, estaremos trabajando con almacenamiento caliente. Esto significa que los datos almacenados son considerados "activos" y consideramos que se acceden con frecuencia. Debido a que se trata de asistencias y participaciones de los alumnos, es indispensable tener acceso inmediato y en tiempo real a esta información, especialmente para las consultas que podrían ser realizadas por los administradores. Este tipo de almacenamiento garantiza que estos datos estén siempre listos para ser recuperados rápidamente, optimizando la eficiencia del sistema.

Arquitectura de datos

A continuación se muestra la arquitectura que se utilizará para el sistema, las herramientas que se utilizarán para cada parte y como se comunican entre ellas. La arquitectura se compone de dos partes principales, donde el frontend es lo que ve el usuario final y con lo que interactúa y el backend es el encargado de comunicarse con todos los servicios y hacer que la página sea funcional.

Sistema de Detección de Asistencia y Participación



Como se puede observar, se cuenta con los siguientes elementos:

1. Frontend:
 - a. Cliente: El cliente es el navegador web desde donde se consume la aplicación web de administración y acceso a la información de la plataforma en general.
 - b. App Frontend: Esta es la aplicación web a la que accede a la información, reportes y administración general de la plataforma.
2. Backend:
 - a. App Backend: Esta es la aplicación que gestiona las conexiones entre los elementos del Frontend con los elementos del backend a través de un modelo de APIs (Application Programming Interfaces) las cuales permiten acceder, generar y modificar información mediante los endpoints de las mismas.
 - b. ORM: Este módulo tiene la función de simplificar la conexión y ejecución de queries entre nuestros endpoints en el backend con la base de datos.

- c. Base de Datos: En esta base de datos no relacional se almacena la información de los usuarios, cursos, asistencias y participaciones.
- d. Modelo de Detección: Este módulo es el responsable de la detección de video e imágenes, procesamiento de las mismas, registro del rostro de nuevos estudiantes así como de la detección de rostros para la toma de asistencias y del reconocimiento de poses para el registro de las participaciones.

Dentro del diagrama también se puede observar donde están hospedados los módulos (AWS Lightsail y Servidores Locales) así como también el tipo de conexiones y protocolos que se utilizan para la comunicación y transmisión de datos.

Tipo de datos a utilizar

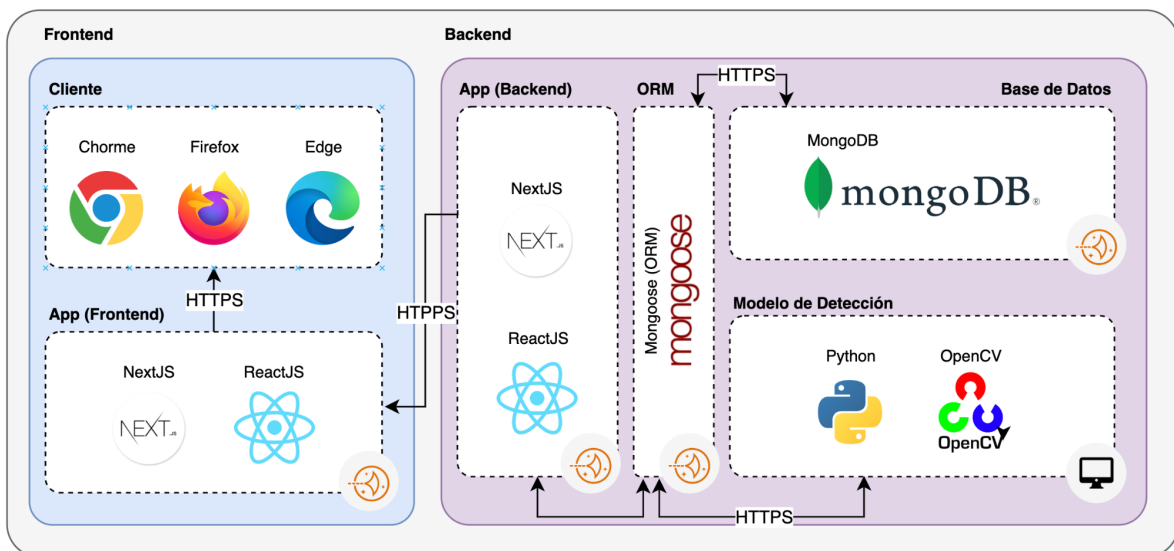
Los datos que se están almacenando en este proyecto están relacionados con la información de las asistencias y participaciones de los alumnos en los diferentes cursos. Al estar utilizando MongoDB para almacenar estos datos, se está trabajando con un almacenamiento de **block storage**, el cual consiste en el almacenamiento de datos en bloques fijos en un entorno de disco. Cada bloque actúa como un contenedor individual de datos y tiene su propia dirección, lo que permite un acceso rápido y eficiente a partes específicas de los datos, sin tener que leer todo el conjunto.

Además, estaremos trabajando con almacenamiento caliente. Esto significa que los datos almacenados son considerados "activos" y consideramos que se acceden con frecuencia. Debido a que se trata de asistencias y participaciones de los alumnos, es indispensable tener acceso inmediato y en tiempo real a esta información, especialmente para las consultas que podrían ser realizadas por los administradores. Este tipo de almacenamiento garantiza que estos datos estén siempre listos para ser recuperados rápidamente, optimizando la eficiencia del sistema.

Arquitectura de datos

A continuación se muestra la arquitectura que se utilizará para el sistema, las herramientas que se utilizarán para cada parte y como se comunican entre ellas. La arquitectura se compone de dos partes principales, donde el frontend es lo que ve el usuario final y con lo que interactúa y el backend es el encargado de comunicarse con todos los servicios y hacer que la página sea funcional.

Sistema de Detección de Asistencia y Participación



Como se puede observar, se cuenta con los siguientes elementos:

1. Frontend:
 - a. Cliente: El cliente es el navegador web desde donde se consume la aplicación web de administración y acceso a la información de la plataforma en general.
 - b. App Frontend: Esta es la aplicación web a la que accede a la información, reportes y administración general de la plataforma.
2. Backend:
 - a. App Backend: Esta es la aplicación que gestiona las conexiones entre los elementos del Frontend con los elementos del backend a través de un modelo de APIs (Application Programming Interfaces) las cuales permiten acceder, generar y modificar información mediante los endpoints de las mismas.
 - b. ORM: Este módulo tiene la función de simplificar la conexión y ejecución de queries entre nuestros endpoints en el backend con la base de datos.

- c. Base de Datos: En esta base de datos no relacional se almacena la información de los usuarios, cursos, asistencias y participaciones.
- d. Modelo de Detección: Este módulo es el responsable de la detección de video e imágenes, procesamiento de las mismas, registro del rostro de nuevos estudiantes así como de la detección de rostros para la toma de asistencias y del reconocimiento de poses para el registro de las participaciones.

Dentro del diagrama también se puede observar donde están hospedados los módulos (AWS Lightsail y Servidores Locales) así como también el tipo de conexiones y protocolos que se utilizan para la comunicación y transmisión de datos.

Aplicación de Big Data y utilización de datos

Por ahora en el proyecto no se tiene planeado utilizar una gran cantidad de datos, ya que el propósito es solamente crear una prueba de concepto y que se demuestre si es un proyecto viable de desarrollar o no. De todas formas, en caso de desarrollar este proyecto como un producto final, se debe considerar el uso de grandes cantidades de datos.

Primero, el hecho de tener a todos los alumnos, cursos y profesores hace que sea necesario almacenar información en la base de datos. Para permitir un escalamiento sencillo, se utiliza la base de datos en MongoDB, descrita más arriba. Por otro lado, al trabajar con modelos de inteligencia artificial que estarán corriendo en dispositivos externos, es necesario que tengan acceso a una base de datos en la que no sea tan importante como guardar los datos, si no que solo los guarden (no relacional).

Para hacer la detección de los alumnos y profesores, es necesario almacenar los encodings de las caras de cada quien, el sistema puede requerir de mucho almacenamiento al escalar el sistema.

Como ya se mencionó, el propósito del proyecto es tener la prueba de concepto del sistema. Es por esta razón que se están utilizando herramientas pre entrenadas en lugar de hacer modelos de reconocimiento desde 0. El generar un set de datos desde 0 para entrenar y probar los modelos de detección de caras y detección de participación puede ser muy caro y muy tardado y es más viable adaptar tecnologías que ya existen para que cumplan los requerimientos de nuestro sistema.