



# Tecnológico de Monterrey

*Instituto Tecnológico y de Estudios Superiores de Monterrey*

**Momento de Retroalimentación: Reto Documentación**

## **TC3007C.501 Inteligencia Artificial Avanzada para la Ciencia de Datos II**

**Profesores:**

*Iván Mauricio Amaya Contreras*

*Blanca Rosa Ruiz Hernández*

*Félix Ricardo Botello Urrutia*

*Edgar Covantes Osuna*

*Felipe Castillo Rendón*

*Hugo Terashima Marín*

### **Equipo 2**

**Integrantes:**

*Luis Ángel Guzmán Iribe – A01741757*

*Julian Lawrence Gil Soares – A00832272*

*Alberto H Orozco Ramos – A00831719*

**1 de Noviembre del 2023**

# Índice

<b>Gantt: Plan de Proyecto - Gantt.mpp.....</b>	<b>2</b>
<b>Metodología:.....</b>	<b>2</b>
<i>Fase 1: Planeación.....</i>	<i>2</i>
1. Definición de Equipo.....	2
2. Planteamiento a resolver.....	2
3. Delimitación de las herramientas computacionales a utilizar.....	3
4. Definición y elaboración del documento de requerimientos.....	3
5. Establecimiento de acuerdo en el Memorándum de Entendimiento.....	3
6. Precisar la metodología a implementar durante el desarrollo del proyecto.....	3
<i>Fase 2: Inicialización.....</i>	<i>3</i>
1. Establecimiento del dataset y los recursos para obtener los datos (emulación o vídeos).....	4
2. Construcción de los primeros prototipos de reconocimiento facial.....	4
3. Construcción de los primeros prototipos de los modelos de reconocimiento de movimiento.....	4
4. Desarrollo e implementación de los primeros componentes de la plataforma web.....	4
5. Extracción, limpieza, separación y carga de los datos con modelo de almacenamiento.....	4
<i>Fase 3: Desarrollo.....</i>	<i>5</i>
1. Elaboración e implementación de los primeros modelos de ML con los datos definidos.....	5
2. Desarrollo, mejoramiento y aplicación de los modelos de Deep Learning (Pose y Facial).....	5
3. Conclusión de Desarrollo de Interfaz Web.....	5
4. Implementación de modelo con la infraestructura web.....	6
5. Establecimiento y aplicación de las métricas de desempeño para comparar los modelos implementados.....	6
<i>Fase 4: Cierre.....</i>	<i>6</i>
1. Entrega final y presentación de la resolución del reto.....	6

## Gantt: [Plan de Proyecto - Gantt.mpp](#)

### Metodología:

#### Fase 1: Planeación

Durante la fase inicial de nuestro reto tuvimos una junta con el socio formador NDS, durante esta junta definimos la problemática que vamos a resolver a lo largo del desarrollo del reto. Se nos pidió elaborar un sistema automatizado el cual pueda llevar un control de la asistencia de cada alumno en los cursos, así como un sistema con el cual se pueda medir la participación de los alumnos dentro del aula. Después de definir la problemática nos reunimos en equipo para llevar a cabo los pasos necesarios antes de comenzar el desarrollo. En primera instancia, establecimos las herramientas a utilizar; para la parte de web decidimos usar VueJS, FastAPI y MongoDB, adicionalmente decidimos utilizar AWS para el alojamiento de nuestra aplicación web. Elegimos utilizar VueJS principalmente por familiaridad con este framework de desarrollo web. Como nuestros algoritmos para participación y reconocimiento serán implementados utilizando el lenguaje de programación Python, nuestras mejores opciones para el backend eran frameworks que utilicen este mismo lenguaje como base con el fin de generar una conexión amena entre los modelos, la base de datos, el backend y el frontend de la aplicación. Lo anterior nos hizo plantearnos frameworks como Django, FastAPI y Flask. Algo crucial para nuestro proyecto es que nuestros algoritmos pueden funcionar de manera asincrónica y debido a que FastAPI posee una implementación sencilla de dichas funciones, optamos por utilizar este framework para el backend. En cuanto a la medición de participación dentro del aula, decidimos utilizar librerías o benchmarks de Python que nos apoyaran para construir el sistema de forma sencilla, esto por recomendación de los profesores auxiliares del reto. Algunas de estas bibliotecas son: *OpenCV*, *MediaPipe*, *face\_recognition* y *YOLOv5*. Una de las razones principales por las que decidimos utilizar estas bibliotecas es debido a que cuentan con modelos pre-entrenados. Como no tenemos suficiente tiempo ni los recursos necesarios para conseguir los datos necesarios, además, carecemos del poder computacional para entrenar los modelos de forma apropiada, esta fue la medida más apropiada y factible de optar por estas opciones. De igual forma, los profesores del reto nos apoyaron con esta idea, siendo que la naturaleza de los algoritmos de

Deep Learning es muy exhaustiva computacionalmente y requiere de mucho tiempo para entrenar, validar y probar su rendimiento y los resultados proporcionados. Por último, debido a que no contamos con datos reales con los que podemos probar nuestro sistema, implementamos una animación de un salón con alumnos utilizando Unity. Una vez definidas las herramientas dividimos las actividades para llevar a cabo el proyecto, así como la distribución de estas entre los integrantes del equipo. Esta fase se llevó a cabo en la primera semana.

### **Descripción de actividades:**

#### **1. Definición de Equipo**

Organizar y construir los equipos para el reto.

#### **2. Planteamiento a resolver**

Definir, entender y analizar la problemática a resolver.

#### **3. Delimitación de las herramientas computacionales a utilizar**

Establecer las herramientas a utilizar basándose en el alcance, presupuesto, recursos y tiempo estipulado para generar la solución de la problemática.

#### **4. Definición y elaboración del documento de requerimientos**

Definición y levantamiento de los requerimientos funcionales que debe cumplir el software para su correcto funcionamiento y cumplimiento de los objetivos que busca cubrir el socio formador con el producto final.

- **Recolección de Datos:** El sistema deberá ser capaz de extraer datos de videos estáticos o de un feed proveniente de una cámara web, garantizando una calidad de imagen suficiente para el reconocimiento facial.
- **Almacenamiento Seguro de Datos:** La plataforma debe asegurar la integridad y confidencialidad de la información mediante medidas de seguridad robustas, como cifrado de datos y acceso restringido.
- **Reconocimiento Facial y Almacenamiento de Alumnos:** Se requiere la capacidad de reconocer y diferenciar rostros de estudiantes, asignándoles un identificador único. La información debe ser almacenada de manera segura y en conformidad con las regulaciones de privacidad.
- **Registro de Asistencia:** Implementar una función que detecte la presencia de un alumno en un aula virtual o física, registrando de manera precisa su asistencia.

- **Detección de Participación:** Desarrollar una función que evalúe la participación de los alumnos, basándose en criterios personalizables definidos por cada equipo o profesor.
- **Roles y Permisos:** Establecer tres roles distintos (alumno, profesor y administrador) con permisos específicos. Esto incluye limitar o ampliar el acceso a funciones dentro de la plataforma, según el rol asignado.
- **Visualización de Estadísticas:** Incorporar gráficos y visualizaciones que muestran estadísticas detalladas sobre la participación y asistencia de los alumnos, permitiendo un análisis eficiente para el cuerpo docente.
- **Gestión de Clases:** Ofrecer funcionalidades para la creación, modificación y eliminación de clases. Esto incluye la posibilidad de ajustar el nombre de la clase, el horario y la ubicación del aula.
- **Infraestructura Escalable:** Diseñar el sistema con una arquitectura escalable que pueda adaptarse al crecimiento de usuarios y datos, garantizando un rendimiento óptimo incluso en situaciones de carga elevada.

## **5. Establecimiento de acuerdo en el Memorándum de Entendimiento**

Redacción de un [documento](#) en el cual se explique brevemente los acuerdos entre el equipo de trabajo y el socio formador, especificando qué es lo que se espera generar.

## **6. Precisar la metodología a implementar durante el desarrollo del proyecto**

Se establece la metodología a seguir paso a paso para asegurar un buen desarrollo del proyecto, organización de actividades y roles de equipo para trabajar.

## **Fase 2: Inicialización**

En la fase de inicialización empezamos a implementar los prototipos de cada componente que se va a requerir para completar el reto, estas son: La aplicación web, una base de datos en la que podemos almacenar nuestros datos, reconocimiento facial, detección de objetos y detección de poses. En esta fase principalmente encontramos soluciones iniciales y creamos prototipos para tener una base de cada componente que vamos a necesitar.

### **1. Establecimiento del dataset y los recursos para obtener los datos (emulación o vídeos)**

Definición del tipo de datos a utilizar y manejar en nuestro proyecto (videos, imágenes, etc.)

### **2. Construcción de los primeros prototipos de reconocimiento facial**

Codificación del primer modelo de reconocimiento facial para evaluar la factibilidad del proyecto, el poder de cómputo mínimo necesario para ejecutar dicha tarea y evaluación de desempeño de detección y reconocimiento para la asistencia. El primer modelo de reconocimiento facial nos sirvió como una buena base para nuestro modelo final. Aprendimos que utilizando un modelo pre-entrenado para detección de rostro podemos implementar un algoritmo que detecte y luego solo es cuestión de reconocer los rostros contra imágenes que nosotros enviamos al algoritmo.

### **3. Construcción de los primeros prototipos de los modelos de reconocimiento de movimiento**

Codificación del primer modelo de reconocimiento de pose para evaluar la factibilidad del proyecto, el poder de cómputo mínimo necesario para ejecutar dicha tarea y evaluación de desempeño de detección y reconocimiento de pose para la participación. En la fase de prototipos pasamos por muchas iteraciones que no cumplían con las funciones que requerimos, pero aprendimos que integrando las bibliotecas de YOLOv5 y media pipe podemos cumplir nuestros requisitos. La biblioteca de YOLOv5 nos permite hacer reconocimiento de objetos, pero no cuenta con ningún modelo pre-entrenado que nos permite detectar la participación. En el caso de mediapipe, podemos generar poses sobre una persona para así implementar un algoritmo que mida participación si se encuentra una pose específica en una persona. El problema que encontramos con esta biblioteca es que cuando hay más de una persona dentro del video, no se agregan poses correctamente. Pero si combinamos los algoritmos podemos usar YOLOv5 para detectar dónde están todas las personas, y luego agregarles una pose de media-pipe manualmente.

### **4. Desarrollo e implementación de los primeros componentes de la plataforma web**

Diseño y creación de la infraestructura web a utilizar para llevar el seguimiento de las clases, involucrando datos como las asistencias y participaciones de cada

uno de los alumnos. El sistema consta de 3 componentes principales, una interfaz de usuario Web, un servidor backend que se encarga del procesamiento de información que se comunica a través de credenciales únicas con los 2 últimos componentes, un bucket de S3 para almacenar las imágenes de los estudiantes y la base de datos en MongoDB; la imagen del estudiante se asocia por medio de un ID único con los datos del mismo almacenados en MongoDB.

## **5. Extracción, limpieza, separación y carga de los datos con modelo de almacenamiento**

La preparación y tratamiento de los datos conlleva suprimir o ajustar dimensiones de algunas imágenes que para el algoritmo puedan llegar a no ser muy claras de interpretar, además, esto conlleva organizar los datos en distintos directorios, con la finalidad de que tanto los datos de entrada como los de pruebas se encuentren distribuidos correctamente por dichos agrupamientos previo a su uso en los modelos de reconocimiento facial y detección de pose. Esto involucra asegurarse de que los datos son fiables, coherentes y que cumplen con ciertas características y requisitos, tales como que la calidad de las imágenes es clara, por lo menos hay un rostro visible de frente a la cámara dentro de la imagen y que dicha foto coincida con el de un estudiante registrado en el curso. En nuestro caso, las imágenes de los estudiantes son procesadas por un algoritmo, no se encuentran corruptos debido a que restringimos los formatos de imagen permitidos, poseen una resolución legible y se enfoca el rostro de frente. Los videos de las clases tienen buena resolución, no se encuentran corruptos y se tiene una vista general de todo el salón de clase para captar a los estudiantes y sus rostros.

## **Fase 3: Desarrollo**

En la fase de desarrollo nuestro equipo dividió el trabajo (en base a lo indicado en el Diagrama de Gantt mencionado) para poder desarrollar todos los componentes en paralelo. El primer paso fue desarrollar la aplicación web y el reconocimiento facial, luego implementamos el sistema de asistencia y comenzamos el desarrollo del sistema de participación. Una vez que completamos los componentes individuales empezamos a integrarlos en la aplicación web. Esto se implementa por medio de Fast API. Este backend, como se mencionó anteriormente, se eligió por su sencillez, facilidad y por que utiliza Python como lenguaje principal, por lo que implementar

endpoints que activen a los modelos es muy sencillo, solo hace falta conectar dicho backend con las credenciales de la base de datos (MongoDB), el bucket de S3 (AWS) y el frontend (VueJS). Cada componente conlleva retos distintos así que los dividimos para tener a alguien que se especializa en cada componente.

### **Descripción de actividades:**

#### **1. Elaboración e implementación de los primeros modelos de ML con los datos definidos**

Se hace uso de los modelos de reconocimiento facial (face\_recognition) y detección de pose (YOLOv5, Mediapipe, face\_recognition) con los datos limpios, con el fin de corroborar su buen funcionamiento.

Esto se hace por medio de pruebas, debido a que son modelos pre entrenados, no podemos entrenarlos y/o validarlos, así que optamos por evaluar su desempeño por medio de la observación, es decir, por el lado del reconocimiento facial sí reconoce rostros dibujando un recuadro, es capaz de reconocer a las personas correctamente y demostrarlo en pantalla con alguna etiqueta cerca del recuadro dibujado, o bien con el modelo de pose, si es que se adapta bien a la pose de la persona, si existe mucha o poca confusión en la pose, si es capaz de reconocer cuando los puntos o “landmarks” que conforman a los brazos se encuentran por encima de los hombros para toma de participación, entre otros detalles.

#### **2. Desarrollo, mejoramiento y aplicación de los modelos de Deep Learning (Pose y Facial)**

En base a los resultados obtenidos, se deberán ajustar los parámetros correspondientes, como la tolerancia en el modelo de reconocimiento facial para distinguir entre personas o la precisión que se desea utilizar en el modelo de pose. Lo anterior se hace con el fin de mejorar su rendimiento, precisión e interpretación tanto de imágenes como de videos, previo a su implementación en conjunto con los 2 modelos funcionando.

#### **3. Conclusión de Desarrollo de Interfaz Web**

Se termina el desarrollo de la plataforma, cumpliendo con todos los requerimientos estipulados para la misma, mostrando datos relevantes para un profesor (promedio de asistencias y participaciones), incluyendo opciones para subir fotografías (perfil de estudiante), y grabaciones de sesiones de clase (perfil de docente).



#### **4. Implementación de modelo con la infraestructura web**

Integración de todas las herramientas y componentes desarrollados en una misma sola aplicación (frontend, backend, MongoDB, AWS, Fast API, modelos pre-entrenados de Deep Learning).

#### **5. Establecimiento y aplicación de las métricas de desempeño para comparar los modelos implementados**

Definición de las métricas a aplicar para evaluar los modelos implementados de reconocimiento facial y de pose. En nuestro caso al ser modelos pre entrenados, no podemos evaluar con métricas como matrices de confusión, o por precisión o exactitud, simplemente podemos evaluar que tan buenos son los modelos con pruebas como por ejemplo con el modelo de reconocimiento facial, ¿a cuántas personas es capaz de reconocer al mismo tiempo? ¿coincide la persona con el nombre que le otorga el algoritmo? ¿qué tan bien lo hace en imágenes? ¿qué tan bien lo hace con videos? ¿dentro de los videos no se confunde a lo largo del paso de los frames? Lo mismo podemos aplicar al modelo de pose, ¿qué tan bien se ajusta a la pose de las personas en la imagen? ¿somos capaces de aprovechar los puntos o “landmarks” para detectar un brazo alzado? ¿podemos lidiar con más de un modelo al mismo tiempo? Este tipo de cuestiones son consideradas a lo largo de la implementación de los modelos para corroborar su correcto funcionamiento.

#### **Fase 4: Cierre**

Durante esta última etapa nos encargamos de realizar pruebas para asegurarnos que nuestro proyecto cumple con las necesidades del socio formador, integraremos todos los componentes y realizaremos documentación de nuestro proyecto. Para finalizar el proyecto realizaremos una presentación en la que detallamos la funcionalidad de nuestro proyecto.

#### **Descripción de actividades:**

##### **1. Entrega final y presentación de la resolución del reto**

Se lleva a cabo la presentación final del producto generado al socio formador durante horas de clase junto a docentes y el resto de los equipos.