



**FACULTAD
DE INGENIERIA**

Universidad de Buenos Aires

People Behavior Tracking (PBT)

Autor:

Hernán Contigiani

Director:

Urbano Pasquinel (Globant)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 5 de marzo de 2021 y el 23 de abril de 2021.*

Índice

Registros de cambios	3
Acta de constitución del proyecto.	4
Descripción técnica-conceptual del proyecto a realizar	5
Identificación y análisis de los interesados.	6
1. Propósito del proyecto.	7
2. Alcance del proyecto	7
3. Supuestos del proyecto.	8
4. Requerimientos	9
Historias de usuarios (<i>Product backlog</i>)	10
5. Entregables principales del proyecto	11
6. Desglose del trabajo en tareas	12
7. Diagrama de Activity On Node	13
8. Diagrama de Gantt.	14
9. Matriz de uso de recursos de materiales	15
10. Presupuesto detallado del proyecto	16
11. Matriz de asignación de responsabilidades	16
12. Gestión de riesgos.	17
13. Gestión de la calidad	19
14. Comunicación del proyecto	21
15. Gestión de compras.	21
16. Seguimiento y control.	22
17. Procesos de cierre.	22

Registros de cambios

Revisión	1c—Detalles de los cambios realizados	Fecha
1.0	Creación del documento	15/03/2021
1.1	Avances de la planificación	17/03/2021
1.2	Modificaciones basadas en las devoluciones Historias de usuario	22/03/2021
1.3	Modificaciones basadas en las devoluciones Gestión del tiempo y los requerimientos	30/03/2021
1.4	Completar los últimos puntos de la planificación	06/04/2021
1.5	Agregar fecha de finalización y corrección de typos	11/04/2021
1.6	Modificaciones basadas en las devoluciones	21/04/2021

Acta de constitución del proyecto

Buenos Aires, 5 de marzo de 2021

Por medio de la presente se acuerda con el Ing. Hernán Contigiani que su Trabajo Final de la Carrera de Especialización en Inteligencia Artificial se titulará “People Behavior Tracking (PBT)”, consistirá esencialmente en utilizar computer vision y deep learning para obtener información sobre el comportamiento de personas/clientes dentro de un recinto, y tendrá un presupuesto preliminar estimado de 748 hs de trabajo, con fecha de inicio 5 de marzo de 2021 y fecha de finalización agosto de 2021.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg
Director posgrado FIUBA

Technical Leader
Globant

Urbano Pasquinel
Director del Trabajo Final

Descripción técnica-conceptual del proyecto a realizar

El sistema de monitoreo de comportamiento de personas (PBT) tiene como objetivo estudiar los movimientos que realiza una persona al ingresar a un espacio, a fin de obtener métricas sobre los lugares del local que visitó, cuánto tiempo permaneció en el recinto y si transitó por alguna zona de interés predefinida.

Para poder cumplir con el objetivo de estudiar el comportamiento de las personas dentro de un espacio es necesario poder detectar a las distintas personas, realizar un seguimiento de cada una y poder identificarlas aún cuando desaparecen del espacio de visión por unos segundos.

Para poder alcanzar el objetivo planteado se debe construir un pipeline de inteligencia artificial como se observa en la Figura 1, el cual está conformado por un detector de personas, un tracker (seguidor) y un sistema que permita extraer características (embeddings) que luego el Engine utilizará para poder identificar personas que hayan salido del rango de visión por oclusiones o fuera de imagen.

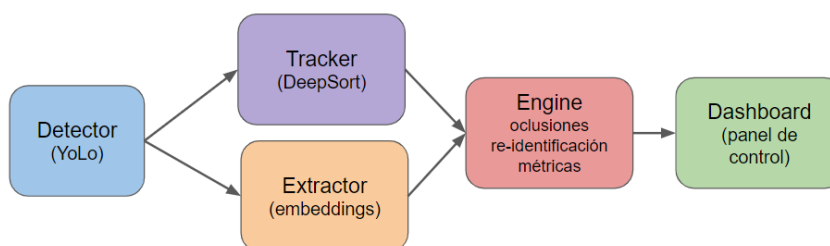


Figura 1. Diagrama en bloques del sistema

El objetivo principal del Engine es identificar a cada persona con un “id” único y analizar sus movimientos dentro del recinto, como también debe resolver las siguientes problemáticas:

- Resolver que una persona salga y entre del rango de visión, y en dicho caso, el sistema la debe identificar como la misma persona.
- Resolver las oclusiones que puedan llegar a ocurrir (personas que tapan a otras personas en el rango de visión).
- Mantener el id correcto entre las personas y evitar que los “id” se intercambien.

Toda la información recolectada por el Engine será enviada a una aplicación web desarrollada en Python, la cual mostrará un dashboard con toda la información recolectada al momento. El Engine para poder realizar todo el labor mencionado necesita como entrada las personas detectadas en un video/imagen (la bounding box de cada una), un "id" tentativo entregado por el Tracker, y características (embeddings) entregadas por un extractor. Para ello ya se estuvo trabajando y ensayando los siguientes modelos pre-entrenados:

- **Detector:** Se utilizará el modelo pre-entrenado "Yolo" como detector por excelencia para detectar personas en una imagen/video. Se utilizará la versión completa de Yolo, a fin de obtener la mayor precisión posible.
- **Tracker:** Se utilizará el modelo pre-entrenado "DeepSort" como tracker por excelencia para asignar un primer "id" tentativo utilizando flujo óptimo, técnicas de seguimiento de patrones (Mean Shift) y un clasificador básico para resolver oclusiones momentáneas.
- **Extractor de características:** Se ensayarán diferentes alternativas como extractor de atributos, extractor de vectores de personas, clasificadores de imágenes entrenados con datasets de personas, etc. Sea cual sea el modelo definitivo que se utilice, el objetivo es obtener un vector de características de cada persona detectada que permita al Engine identificar a las personas en el recinto.

NOTA: Las redes generadores de vectores (extractor) ya se encuentran pre entrenadas con datasets de personas y los modelos/pesos están disponibles en internet. Por una cuestión académica y reforzar este trabajo se realizará el entrenamiento fine-tuning de la red seleccionada como extractor, para luego comparar contra el modelo utilizado disponible en internet. Los datasets más utilizados para este propósito son el Market 1051 y DukeMTMC-reID.

Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Auspiciante	Empresa EEUU	(confidencial)	
Cliente	Technical Leader	Globant	
Impulsor			
Responsable	Hernán Contigiani	FIUBA	Alumno
Colaboradores			
Orientador	Urbano Pasquinel	Globant	Director Trabajo final
Equipo	(confidencial)	Globant	IoT Engineer
Opositores			
Usuario final			

Nota: Por motivos de confidencialidad no puedo mencionar el nombre del auspiciante ni de los miembros del equipo.

1. Propósito del proyecto

El propósito de este proyecto es obtener métricas de los movimientos que realiza una persona al ingresar a un espacio, a fin de obtener los siguientes indicadores:

- Determinar cuántas personas se encuentran en el espacio de interés.
- Determinar zonas de interés en el espacio y determinar cuántas y cuales personas transitaron por este. Las zonas de interés se dibujarán con polígonos en la imagen que representará al espacio de interés.
- Determinar cuánto tiempo las personas estuvieron dentro del espacio y las diferentes zonas de interés.
- Determinar cuando una persona abandona el espacio.

2. Alcance del proyecto

Todas las pruebas se realizarán sobre videos (mp4) públicos, en donde haya un grupo de personas que permita su estudio de comportamiento. Se utilizarán videos que se acerquen lo mayor posible a un video que podría obtenerse de una cámara de seguridad de una tienda o un espacio. En caso de que la empresa consiga videos que puedan ayudar al problema, estos no podrán compartirse a menos que la empresa lo permita y en todo caso sólo se mostrarán los resultados o métricas alcanzadas en el dashboard. El dashboard **no mostrará el streaming de video** sino una representación del espacio y las métricas sobre el comportamiento de las personas monitoreadas.

Para la elaboración de este proyecto la parte del pipeline correspondiente al Detector + Tracker + Extractor se ejecutará en Colab. Se consumirá un video pregrabado mp4 el cual generará un archivo JSON con todos los datos recolectados del video que serán los inputs del Engine para que este luego los consuma finalizado este proceso. El Engine correrá en una máquina local o dispositivo dedicado utilizando como entrada el archivo de datos, y la aplicación web dashboard en una máquina local o servicio en la nube conectada al Engine en real-time (a medida que el Engine consume el archivo y genera información, los resultados se verán reflejados en la App).

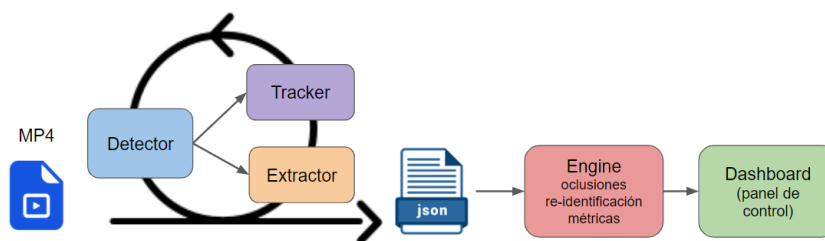


Figura 2. Diagrama de ejecución del sistema

3. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- Todo el proceso que requiera GPU será ejecutado en Colab (entorno virtual gratuito de Google para ejecutar modelos de IA).
- Todo el proceso ejecutado en Colab no será real-time, la salida del sistema será un archivo que consuma luego el Engine.
- En esta fase no se invertirá en hardware, se contará con Colab para el procesamiento pesado en GPU.
- En esta etapa del proyecto no se realizará el deploy del sistema en ningún sistema embebido o servidor, se mantendrá todo en un entorno controlado de desarrollo (Colab + computadora personal).
- Se utilizará diferentes videos que representen un escenario clásico de un local, tienda o recinto.
- En esta fase no será requerido realizar corrección de imagen o deformación de los videos adquiridos, se buscaran videos sin deformaciones.
- No se espera realizar el correcto seguimiento del 100 % de las personas, se establecerá un margen de aceptación para ello.
- No se espera evitar problemas que surjan por movimientos o comportamientos de las personas que salgan de lo normal esperado.

4. Requerimientos

1. Grupo de requerimientos asociados con el pipeline de IA:
 - 1.1. Detectar y seguir personas en un video, excluir otros elementos.
 - 1.2. Generar características que permitan luego re identificar personas perdidas.
 - 1.3. El sistema debe entregar como resultado un archivo con los datos de detección, seguimiento y características en formato JSON.
2. Grupo de requerimientos asociados con el Engine:
 - 2.1. Consumir e interpretar los datos provenientes del pipeline de IA.
 - 2.2. Resolver problemáticas en el seguimiento de las personas utilizando sus características.
 - 2.3. Medir el comportamiento de cada persona con diferentes áreas de interés definidas en la aplicación.
 - 2.4. El sistema debe enviar los resultados en un JSON con toda la información de interés a ser consumida por la aplicación web.
3. Grupo de requerimientos asociados con la aplicación web:
 - 3.1. Mediante una interfaz web poder definir las zonas de interés en el espacio.
 - 3.2. Consumir e interpretar los datos provenientes del Engine.
 - 3.3. Se considerará que una persona es correctamente monitoreada si al menos se mantuvo su seguimiento el 80 % del tiempo que circuló en el recinto.
 - 3.4. Se considerará que el sistema funciona dentro de los parámetros aceptables si entre el 80 % y 100 % de las personas en el video fueron correctamente monitoreadas.
 - 3.5. Poder observar en tiempo real los datos que se obtienen de cada persona y las zonas definidas.
4. Grupo de requerimientos asociados con regulaciones:
 - 4.1. Los videos utilizados no inflijan derechos de privacidad.
 - 4.2. El sistema no asociará las personas en seguimiento con un persona física real.
 - 4.3. Las personas en seguimiento se harán referencia con un número incremental asociado a la base de datos, sin almacenar ningún tipo de información privada.

Historias de usuarios (*Product backlog*)

Roles:

- Usuario - Quien utilizará el sistema desde la aplicación web. Consume los reportes del sistema y lo utiliza para mejorar el layout del recinto (podría ser el área de visuales de la tienda, el área de marketing o el área de diseño de ambientes).
- Product Owner (PO) - El cliente y/o auspiciante del proyecto. Es aquel que vela por la realización del mismo y a su vez porta voz y visión del consumidor final del producto. El PO expone el avance del proyecto a su cartera de clientes a fin de posicionarlo en el mercado, por ello su preocupación e interés está muy enfocado en la experiencia de usuario, el proceso de puesta en marcha del producto en el local y que el proyecto alcance las métricas establecidas a fin de utilizar dicha información como herramienta de negociación para conseguir más inversión.
- Instalador - Quien instala el sistema en la tienda o recinto. El instalador es empleado y entrenado por el auspiciante del proyecto, se vale de la experiencia adquirida en otros productos similares y de las herramientas construidas que lo ayuden a la puesta en marcha del producto en el recinto.

Story points:

- Se realizarán 2 sprints por mes (sprints de 2 semanas).
- Los sprints abarcan entre 40 y 50 horas de trabajo cada uno.
- Se utilizará la escala de 1, 3, 5 y 8 puntos para las historias, siendo 8 puntos una historia que conlleva todo un sprint.
 - Historias de 1 punto: Aquellas de bajo riesgo y rápida implementación principalmente asociadas con cambios cosméticos. Las historias asociadas con 1 punto se estiman que se resolverán en el día.
 - Historias de 3 puntos: Aquellas de bajo riesgo e implementación moderada principalmente asociadas con requerimientos de prioridad moderada. Las historias asociadas con 3 puntos se estiman que consumirán menos de la mitad de la capacidad de todo el sprint.
 - Historias de 5 puntos: Aquellas de riesgo moderado e implementación moderada principalmente asociadas con requerimientos de alta prioridad. Las historias asociadas con 5 puntos se estiman que consumirán la mitad de la capacidad de todo el sprint.
 - Historias de 8 puntos: Aquellas de riesgo alto e implementación compleja principalmente asociadas con requerimientos que requieren una investigación previa o de cuya implementación dependan otras tareas de alta prioridad. Las historias asociadas con 8 puntos se estiman que consumiran la capacidad de todo el sprint.

Historias de usuario:

- Como PO, quiero detectar personas en un video, para poder monitorear su comportamiento. (8p)
- Como PO, quiero re identificar a las personas ocluidas por otras, para mantener el monitoreo aún cuando la persona se pierde de vista por unos segundos. (8p)
- Como PO, quiero conocer el accuracy de monitoreo y seguimiento, para conocer el estado de funcionamiento del sistema.
- Como instalador, quiero una interfaz para dibujar las zonas de interés, para poder realizar el proceso de instalación más rápido. (8p)
- Como instalador, quiero visualizar lo que está detectando el sistema, para poder verificar que la instalación se efectuó con éxito. (5p)
- Como cliente, quiero poder visualizar el monitoreo de las personas en el recinto, para conocer como interactúan con el ambiente. (8p)
- Como cliente, quiero un reporte diario de monitoreo, para conocer la cantidad de personas que ingresaron y el tiempo de permanencia en cada zona. (5p)
- Como cliente, quiero un mapa de calor del recinto, para conocer las zonas en donde hubo mayor interacción. (3p)

5. Entregables principales del proyecto

- Esquema del funcionamiento de cada pieza del pipeline de IA de Colab (código reservado para Globant).
- Esquema del funcionamiento del Engine de seguimiento (código reservado para Globant).
- Aplicación web para interactuar con el sistema y ejecutar los ensayos.
- Informe final

6. Desglose del trabajo en tareas

1. Desarrollo del pipeline IA. (348hs)
 - 1.1. Obtener y analizar videos. (48hs)
 - 1.2. Ensayo de modelos de detección y trackeo. (40hs)
 - 1.3. Ensayo del extractor de características DeepMAR. (40hs)
 - 1.4. Ensayo del extractor de características OsNet. (40hs)
 - 1.5. Integración del extractor de características en el pipeline de detección. (40hs)
 - 1.6. Ensayos de integración profunda del extractor con el tracker. (40hs)
 - 1.7. Integrar al pipeline un modelo para la predicción de poses de la persona. (20hs)
 - 1.8. Armar el dataset para el entrenamiento del modelo extractor con Tensorflow (TF). (30hs)
 - 1.9. Entrenar y ensayar el modelo extractor en TF. (30hs)
 - 1.10. Comparativas entre los modelos extractores entrenados con los modelos pre-entrenados obtenidos en la web. (20hs)
2. Desarrollo del Engine de seguimiento. (160hs)
 - 2.1. Desarrollo del sistema de clustering para la re identificación de personas. (40hs)
 - 2.2. Definir las zonas de interés a partir de un JSON o archivo. (20hs)
 - 2.3. Ubicar las personas en el plano relativo a la imagen, a fin de determinar su ubicación en el recinto (20hs)
 - 2.4. Realizar lógica de detección dentro de zonas. (20hs)
 - 2.5. Desarrollar las APIs para integrar el Engine con dashboard o aplicación web. (20hs)
 - 2.6. Mejorar el accuracy del sistema analizando casos de borde (40hs)
3. Desarrollo del dashboard/web app. (140hs)
 - 3.1. Desarrollar la interfaz de usuario para definir las zonas de interés sobre una imagen (40hs)
 - 3.2. Desarrollar las APIs de comunicación con el Engine. (20hs)
 - 3.3. Desarrollar la interfaz de visualización de las métricas de las personas. (40hs)
 - 3.4. Desarrollar la interfaz de visualización de la representación de las personas en la tienda en tiempo real. (40hs)
4. Presentación del trabajo. (100hs)
 - 4.1. Redacción del informe de avance. (20hs)
 - 4.2. Redacción de las memorias del proyecto. (60hs)
 - 4.3. Preparación de la presentación pública. (20hs)

Cantidad total de horas: (748hs)

7. Diagrama de Activity On Node

En el la Figura 3 se ilustra el diagrama Activity On Node, la ruta crítica es resaltada en cuadros rojos. Aunque solo se cuenta con un recurso humano a lo largo del proyecto, se ejecutan tareas relacionadas en paralelo.

1. Las tareas están expresadas en horas.
2. La duración del camino crítico es de 380 horas.

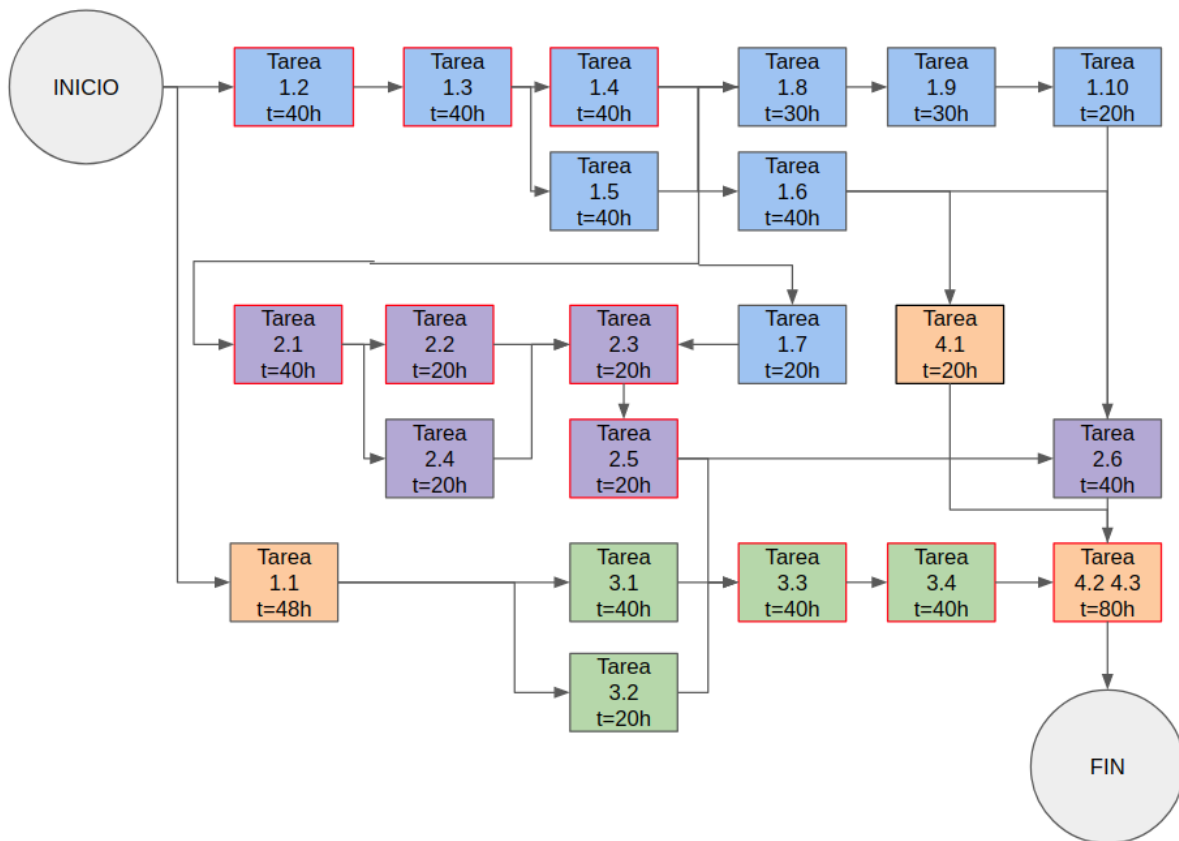


Figura 3. Diagrama Activity On Node

8. Diagrama de Gantt

En la Figura 4 y Figura 5 se ve el diagrama de gantt dividido en la tabla de tareas y la línea de tiempo.

	Name	Start	Finish	Predecessors
1	[48hs] Obtener y analizar videos	12/07/2020	05/28/2021	
2	[40hs] Ensayo de modelos de detección y trackeo	12/03/2020	12/17/2020	
3	[40hs] Ensayo del extractor de características DeepMAR	12/18/2020	01/01/2021	2
4	[40hs] Ensayo del extractor de características OsNet	01/04/2021	01/18/2021	3
5	[40hs] Integración del extractor de características	01/04/2021	01/18/2021	3
6	[40hs] Ensayos de integración profunda del extractor con el tracker	03/04/2021	03/18/2021	4,5
7	[20hs] Integrar al pipeline un modelo para la predicción de poses	03/16/2021	03/31/2021	4,5
8	[30hs] Armar el dataset para el entrenamiento custom del extractor	04/01/2021	04/15/2021	4,5
9	[30hs] Entrenar y ensayar el modelo custom del extractor	04/16/2021	04/23/2021	8
10	[20hs] Comparativas y conclusiones del modelo custom entrenado	04/26/2021	04/30/2021	9
11	[40hs] Desarrollo del sistema de re identificación de personas	02/01/2021	02/12/2021	4,5
12	[20hs] Definir las zonas de interés a partir de un JSON o archivo	02/15/2021	02/26/2021	11
13	[20hs] Ubicar a las personas en el plano relativo a la imagen	04/01/2021	04/09/2021	12,14,7
14	[20hs] Realizar lógica de detección dentro de zonas.	02/15/2021	02/26/2021	11
15	[20hs] Desarrollar las APIs para integrar el Engine con dashboard	05/03/2021	05/14/2021	13
16	[40hs] Mejorar la accuracy del sistema analizando casos de borde	06/01/2021	06/17/2021	1,15,17,18,10
17	[40hs] Desarrollar la interfaz de usuario para definir las zonas	05/03/2021	05/28/2021	13
18	[20hs] Desarrollar las APIs para integrar el dashboard con el Engine	05/03/2021	05/14/2021	13
19	[40hs] Desarrollar la interfaz de visualización de las métricas	06/01/2021	06/23/2021	17,18,15
20	[40hs] Desarrollar la interfaz de visualización de las personas	06/01/2021	06/23/2021	17,15,18
21	[20hs] Redacción del informe de avance	05/31/2021	06/11/2021	6,17
22	[60hs] Redacción de las memorias del proyecto	06/24/2021	07/28/2021	16,21,19,20
23	[20hs] Preparación de la presentación pública	07/29/2021	08/11/2021	22

Figura 4. Tabla de tareas Gantt

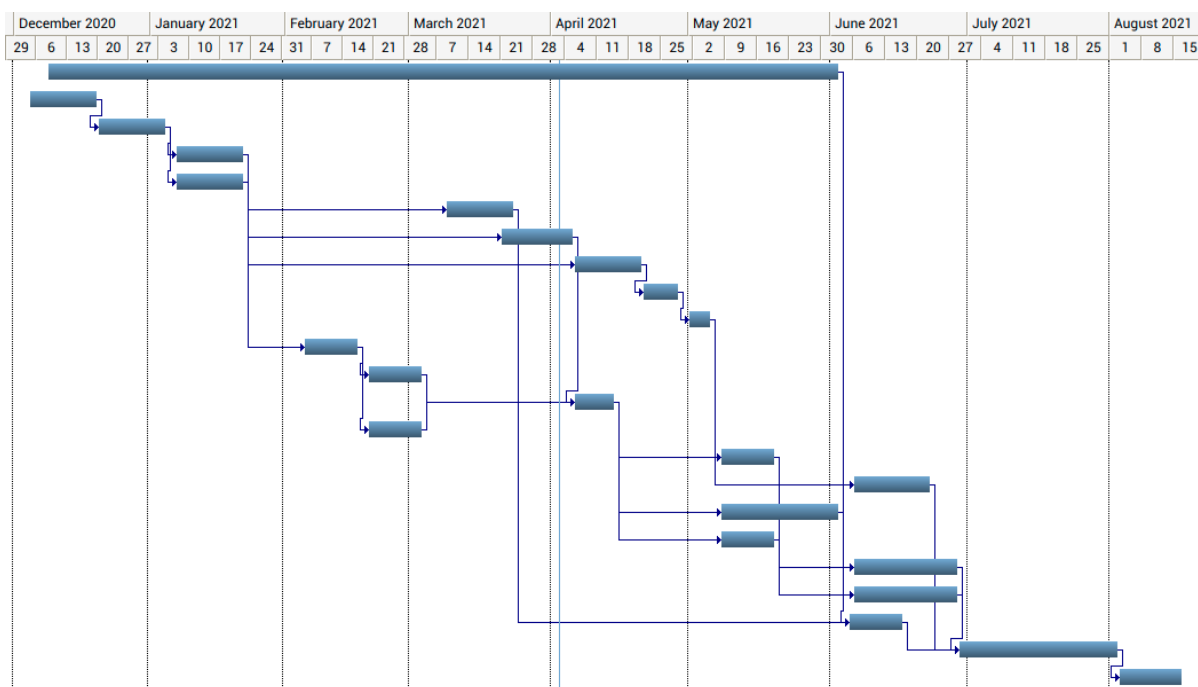


Figura 5. Diagrama de Gantt

9. Matriz de uso de recursos de materiales

Código WBS	Nombre de la tarea	Recursos requeridos (horas)			
		Colab	PC	Youtube	Simulador
1.1	Analizar videos y resultados			30	18
1	Desarrollo Piepline IA	300			
2	Desarrollo del Engine de seguimiento		160		
3	Desarrollo del dashboard / web app		140		

*Simulador será cualquier programa o aplicación que permita obtener videos de muestra fuera de lo que es posible conseguir en canales como youtube, a fin de probar un escenario más cercano al layout de una tienda y poder generar videos más largos. Como posible simulador se podrá utilizar sistemas de renderizado 3D o juegos como el Sims.

10. Presupuesto detallado del proyecto

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Horas de ingeniería	648 horas	\$6 USD	\$3888 USD
Computadora personal	1	\$800 USD	\$800 USD
SUBTOTAL			\$4688 USD
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Licencia del sistema de gestión	1	\$14 USD	\$14 USD
Costo de almacenamiento 100G	1	\$20 USD	\$20 USD
SUBTOTAL			\$34 USD
TOTAL			\$4722 USD

11. Matriz de asignación de responsabilidades

Código WBS	Nombre de la tarea	Listar todos los nombres y roles del proyecto			
		Responsable Hernán Contigiani	Orientador Urbano Pasquinel	Equipo Technical Leader	Cliente Usuario
1	Desarrollo Piepline IA	P	A	I	
2	Desarrollo del Engine	P	I	A	
3	Desarrollo del dashboard	P	I	A	C
4	Presentación del trabajo	P	A	I	

Referencias:

- P = Responsabilidad Primaria
- S = Responsabilidad Secundaria
- A = Aprobación
- I = Informado
- C = Consultado

12. Gestión de riesgos

A lo largo del proyecto se hacen visibles los siguientes riesgos de los cuales se presenta una gestión de los mismos.

Riesgo 1: No disponer del material en video necesario para validar todas las funcionalidades del sistema.

- Severidad (S): Este riesgo tiene una severidad alta porque depende del material en video encontrado para poder validar el pipeline de IA y las funcionalidades que se describen en el proyecto. (10).
- Probabilidad de ocurrencia (O): La probabilidad de este riesgo es alta porque es muy común en proyectos de inteligencia artificial que no se cuenten con tantos datos como los que se deseara. (8)

Riesgo 2: No alcanzar las métricas pautadas para el sistema de seguimiento.

- Severidad (S): Este riesgo tiene una severidad alta porque uno de los factores más importantes del proyecto es la precisión de seguimiento. (8)
- Ocurrencia (O): La probabilidad de este riesgo es baja porque antes de comenzar el proyecto se analizó de que fuera factible alcanzar las métricas pautadas, y en caso de existir una limitación por el layout del local se acotará el área de monitoreo al área efectiva con mayor precisión. (4)

Riesgo 3: No poder re identificar aquellas personas que al salir y entrar en cámara cambian su apariencia o vestimenta.

- Severidad (S): Este riesgo tiene una severidad baja porque parte del margen del sistema contempla estos casos de borde (4).
- Ocurrencia (O): La probabilidad de este riesgo es muy baja porque justo se tiene que dar el caso de que la persona cambie su apariencia fuera de cámara, Mientras la persona esté en el área de visión, el tracker mantendrá el correcto seguimiento a pesar de cambiar su apariencia. (2)

Riesgo 4: No sea posible entrenar el modelo OsNet en Colab.

- Severidad (S): Este riesgo tiene una severidad muy baja porque el entrenamiento del modelo OsNet es unicamente con fin académico. (2)
- Ocurrencia (O): La probabilidad de este riesgo es alta porque los recursos de Colab son limitados. (10)

Riesgo 5: No cumplir con las fechas estimadas de desarrollo del proyecto.

- Severidad (S): Este riesgo tiene una severidad alta porque los recursos dedicados a este proyecto tienen futuras asignaciones concluida esta etapa. (8)
- Ocurrencia (O): La probabilidad de este riesgo es baja porque hasta el momento el proyecto cumplió con todas las fechas importantes. (4)

b) Tabla de gestión de riesgos: (El RPN se calcula como $RPN=S \times O$)

Riesgo	S	O	RPN	S*	O*	RPN*
No disponer del material en video necesario	10	8	80	10	2	20
No alcanzar las métricas pautadas para el sistema de seguimiento	8	4	32	8	3	24
No poder re identificar personas que cambian de apariencia	4	2	8	-	-	-
No sea posible entrenar el modelo OsNet en Colab	2	10	20	-	-	-
No cumplir con las fechas estimadas	8	4	32	8	2	16

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 30.

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

Riesgo 1: Como plan de mitigación se utilizarán simuladores o programas de renderizado 3D para poner a prueba todas las funcionalidades del sistema.

- Severidad (S): La severidad de este riesgo sigue siendo la misma. (10).
- Probabilidad de ocurrencia (O): La probabilidad de este riesgo baja bastante, porque aquello que no pueda ser evaluado con videos reales se llevará a prueba en modelos de renderizado y entornos controlados. (2)

Riesgo 2: Como plan de mitigación está planificada una historia de usuario para contemplar los casos de borde en el sistema y así aumentar la precisión.

- Severidad (S): La severidad de este riesgo sigue siendo la misma. (8).
- Probabilidad de ocurrencia (O): La probabilidad de este riesgo baja muy poco ya que estamos hablando de un trabajo fino, pero se reduce lo suficiente para alcanzar las métricas deseadas. (3)

Riesgo 5: Como plan de mitigación se realizarán reuniones semanales atacar cualquier problema o imprevisto a tiempo.

- Severidad (S): La severidad de este riesgo sigue siendo la misma. (8).
- Probabilidad de ocurrencia (O): La probabilidad de este riesgo baja ya que se llevará un control semanal del avance del proyecto. (2)

13. Gestión de la calidad

- Req #1.1: Detectar y seguir personas en un video, excluir otros elementos.
 - Verificación: consumir un video y verificar visualmente que solamente sean detectadas personas.
 - Validación: Realizar un video post procesado de salida que verifique que solo son detectadas personas en el video.
- Req #1.2: Generar características que permitan luego re identificar personas perdidas.
 - Verificación: Ensayar los diferentes extractores de características propuestos y medir cual de ellos arroja mejores métricas de características. Con dicho extractor calcular si la distancia entre vectores que caracterizan a diferentes personas es mayor a la distancia entre vectores que pertenecen a una misma persona (evaluar que tan diferentes son los vectores de distintas personas vs los vectores de una misma persona).
 - Validación: Realizar un video en donde se vea como el sistema identifica a cada persona en el video utilizando unicamente los vectores obtenidos de estas.
- Req #1.3 y Req #2.1: El sistema debe entregar como resultado un archivo con los datos de detección, seguimiento y características en formato JSON y ser consumido por el Engine.
 - Verificación: Exportar en un archivo JSON la salida del pipeline de IA con los datos de detección, seguimiento y características para luego consumir dicho archivo con el Engine y validar que encuentra correctamente a las personas en el video.
 - Validación: Realizar un video en donde se vea la salida obtenida desde el Engine, validando la detección y el seguimiento de las personas en le video.
- Req #2.3: Medir el comportamiento de cada persona con diferentes áreas de interés definidas en la aplicación.
 - Verificación: Observar los logs de la consola y verificar que la persona fue efectivamente detectada dentro de cada zona definida.
 - Validación: Realizar un video en donde la caja que representa a cada persona se pinte del color de cada zona al ingresar a esta.
- Req #3.3: Se considerará que una persona es correctamente monitoreada si al menos se mantuvo su seguimiento el 80 % del tiempo que circulo en el recinto.
 - Verificación: Observar los logs de las métricas de cada persona y verificar el cumplimiento de dicha métrica.
 - Validación: Realizar un gráfico con el porcentaje de detección de cada persona y verificar el cumplimiento de dicha métrica.

- Req #3.4: Se considerará que el sistema funciona dentro de los parámetros aceptables si entre el 80 % y el 100 % de las personas en el video fueron correctamente monitoreadas.
 - Verificación: Observar los logs de las métricas de cada persona y verificar el cumplimiento de dicha métrica.
 - Validación: Realizar un gráfico con el porcentaje de detección de cada persona y verificar el cumplimiento de dicha métrica.

- Req #4.2 y Req #4.3: El sistema no asociará a las personas en el video con una persona física, se hará referencia a las mismas con un número incremental asociado a la base de datos.
 - Verificación: Observar la base de datos con todos los IDs generados.
 - Validación: Visualizar en la interfaz de la aplicación el seguimiento de las personas por números.

14. Comunicación del proyecto

El plan de comunicación del proyecto es el siguiente:

PLAN DE COMUNICACIÓN DEL PROYECTO					
¿Qué comunicar?	Audiencia	Propósito	Frecuencia	Método de comunicac.	Responsable
Avances de Plan de trabajo	Clase Gestión de Proyectos	Evitar errores y recibir feedback	Semanal	Correo y formulario	Hernán Contigiani
Avances del proyecto	Technical Leader	Avance semanal	Semanal	Video llamada	Hernán Contigiani
Avances de validación y verificación	Director de proyecto	Avance y resultados	Mensual	Video llamada	Hernán Contigiani
Presentación final del proyecto	Jurados y Director	Definición y puesta en común	Única vez	Presentación y memorias del proyecto	Hernán Contigiani

15. Gestión de compras

No es requerido realizar ningún tipo de compra para este proyecto.

16. Seguimiento y control

SEGUIMIENTO DE AVANCE					
Tarea del WBS	Indicador de avance	Frecuencia de reporte	Resp. de seguimiento	Persona a ser informada	Método de comunic.
1	Pipeline IA completo al 54 %, pendiente entrenamiento de OsNet y obtener videos simulados	Semanal	Hernán Contigiani	Technical Leader, Urbano Pasquinel	email
2	Engine completo al 50 %, pendiente el desarrollo de APIs para integrar con la app web y analizar casos de borde	Semanal	Hernán Contigiani	Technical Leader, Urbano Pasquinel	email
3	El desarrollo de la aplicación web no ha comenzado (0 %)	Mensual	Hernán Contigiani	Technical Leader, Urbano Pasquinel	email
4	Presentación del trabajo final (0 %)	Mensual	Hernán Contigiani	Technical Leader, Urbano Pasquinel	email

17. Procesos de cierre

A continuación se describen las pautas que darán cierre al proyecto:

- Reunión por video llamada con el director y el líder del proyecto a fin de analizar las conclusiones del proyecto, retrospectiva y revisión para futuros trabajos.
- Presentación y demostración del proyecto ante jurados mediante diapositivas y/o videos.