



Pronóstico de ventas para toma de decisiones en comercio electrónico usando *Machine Learning*

Autor:

Ing. Jonathan Matías Borda

Director:

Título y Nombre del director (a confirmar) (pertenencia)

Esta planificación fue realizada en el curso de Gestión de proyectos entre el 24 de junio de 2025 y el 12 de agosto de 2025.

Índice

1. Descripción técnica-conceptual del proyecto a realizar.	5
2. Identificación y análisis de los interesados	6
3. Propósito del proyecto	6
4. Alcance del proyecto	7
5. Supuestos del proyecto.	7
6. Product Backlog	8
7. Criterios de aceptación de historias de usuario	9
8. Fases de CRISP-DM	12
9. Desglose del trabajo en tareas	13
6. Requerimientos	16
7. Historias de usuarios (<i>Product backlog</i>).	17
8. Entregables principales del proyecto	18
9. Desglose del trabajo en tareas	18
10. Planificación de Sprints	19
11. Diagrama de Gantt (sprints)	22
12. Normativa y cumplimiento de datos (gobernanza)	23
13. Gestión de riesgos	23
14. Sprint Review	27
15. Sprint Retrospective	28

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	24 de junio de 2025
1	Se completa hasta el punto 5 inclusive	8 de julio de 2025
2	Se completa hasta el punto 9 inclusive	15 de julio de 2025

Acta de constitución del proyecto

Buenos Aires, 24 de junio de 2025

Por medio de la presente se acuerda con el Ing. Jonathan Matías Borda que su Trabajo Final de la Carrera de Especialización en Inteligencia Artificial se titulará “Pronóstico de ventas para toma de decisiones en comercio electrónico usando *Machine Learning*” y consistirá en desarrollar un modelo de inteligencia artificial capaz de pronosticar las ventas de una tienda en línea de la empresa Latech. El trabajo tendrá un presupuesto preliminar estimado de 684 horas y un costo estimado de \$12.825.000, con fecha de inicio el 24 de junio de 2025 y fecha de presentación pública el **21 de junio de 2026 a definir**.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg
Director posgrado FIUBA

Lic. Juan Cruz Bonina
Latech

Título y Nombre del director (a confirmar)
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

La empresa Latech fabrica y comercializa barritas alimenticias en distintos sabores. Como se trata de alimentos, la empresa necesita contar con pronósticos precisos de ventas, que le permitan planificar la producción y evitar tanto faltantes como excedentes de stock.

A partir de esta problemática, se propone desarrollar un modelo de inteligencia artificial capaz de pronosticar las ventas de la tienda en línea de la empresa Latech.

El desafío es que la precisión de la predicción de las ventas sea alta para que el cliente pueda tomar decisiones estratégicas con un alto grado de confianza. A su vez, el cliente desea conocer la evolución de las ventas en el tiempo para poder ajustar la estrategia de marketing y de ventas en función de los resultados.

Se va a utilizar como insumo un conjunto de datos históricos, que abarca aproximadamente dos años, sobre ventas e inversión en publicidad. Estos datos se van a obtener desde Shopify, la plataforma donde el cliente tiene alojada su tienda, con el propósito final de contar con una herramienta que facilite la toma de decisión estratégica basada en la predicción.

La plataforma Shopify está basada en la nube y permita a empresas y particulares crear y gestionar tiendas en línea. Además, esta ofrece una *API* que facilita el acceso a datos como: los productos vendidos por día, los usuarios que realizan la compra, el canal de venta, etc. A través de esta *API* también se obtienen métricas de comportamiento de los usuarios tales como: la cantidad de páginas vistas, cuando agrega un producto al carrito de compra, cuando termina la compra y la tasa de conversión. Adicionalmente, se utiliza una plataforma llamada Triple Whale, especializada en la gestión y análisis de inversión publicitaria que proporciona una *API* donde se obtiene información sobre la inversión realizada por la tienda en sitios como: Google, Facebook, Instagram, Klaviyo, etc.

Para llevar a cabo el proyecto se cuenta con el financiamiento de la empresa Latech y acceso a la *API* de Shopify y a la *API* de Triple Whale. Asimismo, existe un acuerdo de confidencialidad con la empresa, que establece que los datos utilizados y los entregables generados durante el desarrollo del proyecto no podrán ser difundidos públicamente ni compartidos con terceros sin autorización expresa.

En la figura 1 se presenta el diagrama en bloques del sistema. Se observa que las *APIs* externas proveen los datos que alimentan el conjunto de datos del proyecto. Tanto este conjunto de datos como el modelo de predicción se integrarán dentro del sistema del cliente, denominado Inventory Tracker. Como resultado, el sistema generará el pronóstico de ventas.

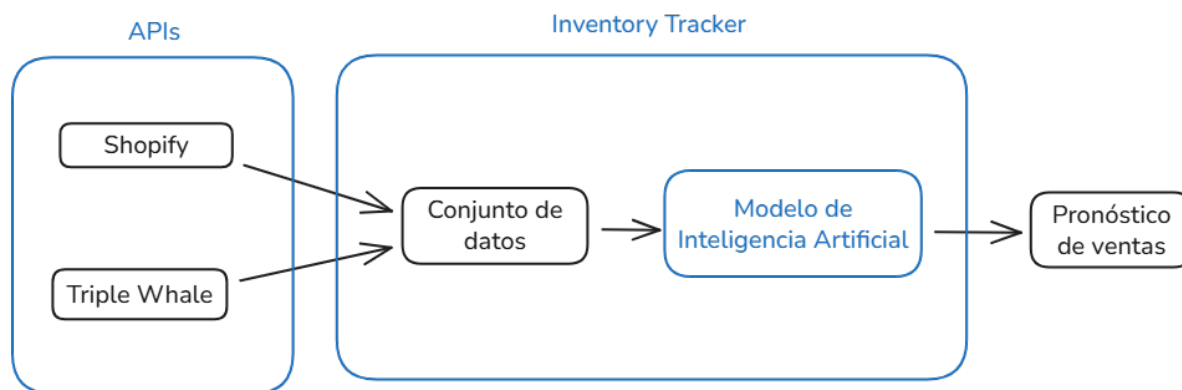


Figura 1. Diagrama en bloques del sistema.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Lic. Juan Cruz Bonina	Latech	CEO
Responsable del Proyecto	Ing. Jonathan Matías Borda	FIUBA	Alumno
Orientador	Título y Nombre del director (a confirmar)	pertenencia	Director del Trabajo Final
Usuario Final	Lic. Juan Willink	Latech	Gerente de Ventas

Cliente: Lic. Juan Cruz Bonina es el CEO de la empresa Latech y quien va a definir los requerimientos. Suele estar disponible en cualquier momento para responder preguntas y brindar comentarios o aclaraciones.

Usuario final: Lic. Juan Willink es el Gerente de Ventas de la empresa Latech y quien va a utilizar el modelo de inteligencia artificial para tomar decisiones estratégicas. Vive en una locación con diferente uso horario, por lo que su disponibilidad es de tiempo parcial.

3. Propósito del proyecto

El propósito de este proyecto es desarrollar un modelo de inteligencia artificial que permita predecir con precisión las ventas del producto alimenticio de la empresa Latech, a partir de datos de ventas y de inversión publicitaria. Esto permitirá a la empresa: anticipar sus necesidades de producción, optimizar la planificación del inventario y tomar decisiones estratégicas basadas en datos. Asimismo, podrá reducir el riesgo de faltantes o excesos de inventario, y mejorar la eficiencia operativa y comercial.

4. Alcance del proyecto

El proyecto incluye:

- Relevamiento y análisis de los datos históricos de ventas de la tienda en línea obtenidos mediante la *API* de Shopify.
- Relevamiento y análisis de los datos históricos de inversión publicitaria provenientes de la *API* de Triple Whale.
- Limpieza, transformación y consolidación de datos provenientes de ambas plataformas.
- Análisis exploratorio de datos (EDA) para identificar patrones, tendencias y relaciones entre las variables.
- Desarrollo y entrenamiento de modelos de inteligencia artificial orientados a la predicción de ventas.
- Evaluación comparativa de distintos modelos para seleccionar el que ofrezca la mejor precisión y capacidad predictiva.
- Implementación de un módulo funcional integrado al sistema Inventory Tracker de la empresa, que permita realizar predicciones de ventas para períodos futuros.
- Documentación técnica del proceso, del modelo elegido y de su uso.
- Entrega de reportes que expliquen los hallazgos y recomendaciones derivadas del análisis.

El presente proyecto no incluye:

- Implementaciones en tiempo real del modelo, es decir, sistemas que actualicen las predicciones de forma instantánea ante cada nuevo dato recibido. Sí se contempla la posibilidad de programar ejecuciones periódicas del modelo (por ejemplo, una vez al día) para actualizar las predicciones de manera regular.
- Garantía de precisión absoluta en las predicciones, ya que la precisión depende de la calidad y estabilidad de los datos futuros y de factores externos no controlables.
- Acciones o recomendaciones específicas sobre estrategias de marketing más allá de lo inferido de los análisis de datos.
- Optimización de procesos internos de producción o logística, salvo en lo que respecta a la estimación de demanda.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- La empresa Latech proveerá acceso completo y continuo a las *APIs* externas de Shopify y Triple Whale para la obtención de datos históricos y actualizados.

- Se contará con acceso al repositorio de código y a la infraestructura necesaria para integrar el módulo de predicción en el sistema existente Inventory Tracker.
- La calidad, integridad y consistencia de los datos obtenidos desde las plataformas externas será suficiente para entrenar y validar los modelos de predicción.
- No se producirán cambios significativos en las políticas de acceso o en la estructura de datos de las *APIs* de Shopify y Triple Whale durante el período de desarrollo.
- No existirán restricciones legales o contractuales que impidan el uso de los datos necesarios para el proyecto.
- Se contará con la disponibilidad de recursos computacionales necesarios para el procesamiento, entrenamiento y validación de los modelos de inteligencia artificial.

6. Product Backlog

El Product Backlog debe organizarse en cuatro **épicas** fundamentales del proyecto. Cada épica debe contener al menos dos historias de usuario que describan funcionalidades clave.

El Product Backlog debe permitir interpretar cómo será el proyecto y su funcionalidad. Se deben indicar claramente las prioridades entre las historias de usuario y si hay alguna opcional.

Las historias de usuario deben ser breves, claras y medibles, expresando el rol, la necesidad y el propósito de cada funcionalidad. También deben tener una prioridad definida para facilitar la planificación de los sprints.

Cada historia de usuario debe incluir una ponderación en *Story Points*, un número entero que representa el tamaño relativo de la historia. El criterio para calcular los Story Points debe indicarse explícitamente.

Las historias deben seguir el formato: “*Como [rol], quiero [tal cosa] para [tal otra cosa]*”.

Las épicas deben estructurarse de la siguiente forma:

1. Épica 1: Integración de fuentes de datos

- 1.1. HU1: “Como analista de datos quiero importar datos de ventas desde Shopify para poder entrenar el modelo de predicción.”
Story points: 5 (complejidad: 2, dificultad: 2, incertidumbre: 1)
- 1.2. HU2: “Como analista de datos quiero importar datos de inversión publicitaria desde Triple Whale para incluir la variable de inversión en el análisis de ventas.”
Story points: 5 (complejidad: 2, dificultad: 2, incertidumbre: 1)
- 1.3. HU3: “Como analista de datos quiero consolidar los datos de Shopify y Triple Whale para tener un único dataset listo para el entrenamiento del modelo.”
Story points: 8 (complejidad: 3, dificultad: 2, incertidumbre: 3)

2. Épica 2: Modelado predictivo

- 2.1. HU4: “Como analista de datos quiero entrenar un modelo que prediga las ventas para poder anticipar la demanda de producción.”
Story points: 13 (complejidad: 5, dificultad: 4, incertidumbre: 4)

3. Épica 3: Visualización y exportación de resultados

3.1. HU5: “Como usuario de Inventory Tracker quiero visualizar las predicciones de ventas dentro del sistema para poder planificar la producción.”

Story points: 8 (complejidad: 3, dificultad: 2, incertidumbre: 3)

3.2. HU6: “Como usuario de Inventory Tracker quiero descargar las predicciones en formato CSV para analizarlas externamente.”

Story points: 3 (complejidad: 1, dificultad: 1, incertidumbre: 1)

4. Épica 4: Monitoreo y alertas del sistema

4.1. HU7: “Como responsable del sistema quiero recibir alertas si hay datos faltantes que puedan afectar significativamente la precisión de las predicciones para tomar medidas al respecto.”

Story points: 5 (complejidad: 2, dificultad: 2, incertidumbre: 1)

4.2. HU8: “Como analista de datos quiero documentar el trabajo para realizar la entrega final del proyecto.”

Story points: 8 (complejidad: 5, dificultad: 3, incertidumbre: 1)

7. Criterios de aceptación de historias de usuario

Los criterios de aceptación deben establecerse para cada historia de usuario, asegurando que se cumplan las condiciones necesarias para que la funcionalidad sea validada correctamente.

Cada historia debe tener criterios medibles, específicos y verificables. Deben permitir validar que se cumple con las necesidades del usuario.

Se estructuran de forma análoga a las épicas del backlog:

■ Épica 1

● Criterios de aceptación HU1

- Se puede establecer conexión con la *API* de Shopify utilizando credenciales válidas.
- Los datos de ventas incluyen al menos: fecha de compra, producto, cantidad, precio y canal de venta.
- El rango de fechas de las ventas a importar se puede configurar.
- Los datos importados se almacenan en una base de datos local o de staging para su procesamiento posterior.
- Se manejan correctamente errores de conexión, autenticación y respuesta vacía.
- Existe un log del proceso de importación que registra la cantidad de registros importados y posibles errores.
- La importación puede ejecutarse manualmente o programarse.

● Criterios de aceptación HU2

- Se puede autenticar correctamente con la *API* de Triple Whale mediante la clave de acceso proporcionada.
- La información importada incluye al menos: fecha, plataforma (Google, Facebook, Instagram, etc.), monto invertido y campaña.

- El rango de fechas para la importación se puede configurar.
- Los datos importados se almacenan en una base de datos estructurada para su uso en el modelo de predicción.
- Se contempla el manejo de errores por fallos en la conexión o respuestas inválidas de la API.
- Se genera un log que detalle la cantidad de registros importados y errores encontrados (si los hubiera).
- La importación puede realizarse de forma programada o manual.
- Se valida que no se dupliquen registros si la importación se ejecuta varias veces para un mismo período.

■ Épica 2

● Criterios de aceptación HU3

- Se genera un único dataset que incluye variables de ventas (provenientes de Shopify) e inversión publicitaria (provenientes de Triple Whale), con correspondencia por fecha.
- Las fechas se alinean correctamente entre ambas fuentes, considerando posibles diferencias de zona horaria.
- En caso de que falten datos para alguna fecha en una de las fuentes, se imputan con valores nulos o definidos por el equipo (por ejemplo, inversión cero).
- El dataset resultante contiene las siguientes columnas mínimas: fecha, ventas diarias, inversión total diaria, desglose por plataforma (si aplica).
- Se valida que no haya duplicados ni registros con formato inválido.
- El proceso de consolidación puede ejecutarse de forma automatizada y repetible.
- El dataset final se guarda en un formato estructurado (por ejemplo, CSV o tabla en base de datos) listo para ser consumido por el modelo.
- Se genera un log de consolidación con la cantidad de registros procesados, errores (si los hay) y fechas incluidas.

● Criterios de aceptación HU4

- Se entrena al menos un modelo de predicción (por ejemplo, regresión lineal, árbol de decisión, red neuronal u otro) utilizando el dataset consolidado de ventas e inversión.
- El conjunto de datos se divide correctamente en entrenamiento y validación (por ejemplo, 80/20 o validación temporal).
- Se calcula y reporta el error de predicción utilizando métricas adecuadas (por ejemplo, MAE, RMSE, MAPE).
- Se selecciona el modelo con mejor desempeño sobre el conjunto de validación.
- El modelo es capaz de generar predicciones futuras en función de datos de entrada recientes.
- El entrenamiento es reproducible: el pipeline puede ejecutarse nuevamente con los mismos resultados (dado un seed o configuración).
- El modelo y los parámetros finales se almacenan en un repositorio o archivo accesible para futuras predicciones.
- Se documenta el proceso de entrenamiento, incluyendo qué variables se usaron, cómo se preprocesaron, y qué algoritmos fueron evaluados.
- Se realiza una validación visual de las predicciones en comparación con los datos reales (por ejemplo, gráfico de ventas reales vs. predichas).

■ Épica 3

- Criterios de aceptación HU5

- Las predicciones de ventas se muestran en la interfaz de Inventory Tracker junto a las fechas correspondientes.
- La visualización incluye tanto las predicciones como los datos históricos de ventas, para permitir la comparación.
- La información está disponible de forma clara y entendible (por ejemplo, mediante gráficos de líneas o tablas).
- El usuario puede seleccionar un rango de fechas para ver predicciones específicas.
- Las predicciones se actualizan automáticamente cuando se entrena un nuevo modelo o se actualizan los datos de entrada.
- La interfaz muestra un indicador de cuándo fue generada la última predicción (timestamp).
- Se indica visualmente que los valores mostrados son predicciones y no datos reales.
- Las predicciones se integran sin errores en la interfaz actual del módulo Inventory Tracker.
- El rendimiento de la interfaz sigue siendo aceptable al mostrar predicciones (por ejemplo, sin demoras perceptibles en la carga).

- Criterios de aceptación HU6

- El sistema muestra un botón o enlace claramente identificado para descargar las predicciones en formato .csv.
- El archivo descargado incluye las fechas y los valores de predicción correspondientes.
- El archivo también incluye, si están disponibles, los datos históricos para comparación (opcional según configuración).
- El formato del .csv es compatible con herramientas comunes de análisis como Excel, Google Sheets o Python (valores separados por comas, codificación UTF-8).
- El nombre del archivo incluye una fecha y hora para identificar cuándo fue generado.
- La descarga se realiza correctamente sin errores en navegadores modernos (Chrome, Firefox, Edge).
- El contenido del archivo refleja exactamente lo que se muestra en la interfaz del módulo de predicción.
- Si no hay predicciones generadas aún, el botón de descarga aparece deshabilitado o muestra un mensaje claro al usuario.

■ Épica 4

- Criterios de aceptación HU7

- El sistema verifica automáticamente la integridad del dataset antes de cada entrenamiento del modelo.
- Si se detectan datos faltantes críticos (por ejemplo, fechas sin ventas o sin inversión publicitaria), se genera una alerta.
- La alerta incluye: Tipo de dato faltante (ventas, inversión, ambos); rango de fechas afectado, nivel de severidad del impacto estimado.

- La alerta se muestra de forma visible en el sistema (por ejemplo, en el dashboard o módulo de predicción).
- También se envía una notificación por correo electrónico al responsable del sistema (si está configurada esta opción).
- El sistema no bloquea el entrenamiento, pero advierte que la precisión puede verse afectada.
- La alerta desaparece solo cuando se completan los datos faltantes o se marca como revisada manualmente.
- Se registra un log de todas las alertas emitidas, accesible desde una sección de administración o monitoreo.
- Criterios de aceptación HU8
 - Se entrega un documento final del proyecto en el formato solicitado por la institución llamado memoria del trabajo final.
 - El documento incluye una descripción clara del problema, objetivos, metodología, resultados y conclusiones.
 - Se documenta adecuadamente el proceso de desarrollo, incluyendo decisiones técnicas, herramientas utilizadas y justificación de enfoques.
 - Se incluyen diagramas de arquitectura, flujo de datos, tablas o visualizaciones necesarias para respaldar el análisis de datos y resultados.
 - El informe cita correctamente todas las fuentes externas y respeta las normas de integridad académica.
 - El documento fue revisado para corregir errores ortográficos y de redacción antes de la entrega.

8. Fases de CRISP-DM

1. **Comprensión del negocio:** El objetivo del proyecto es desarrollar un modelo de inteligencia artificial capaz de pronosticar las ventas de la tienda en línea de la empresa Latech. Esto permitirá anticipar la demanda y planificar la producción de manera eficiente, evitando faltantes o excesos de inventario. El valor agregado de aplicar IA en este contexto es la posibilidad de tomar decisiones estratégicas basadas en datos. Las métricas de éxito del proyecto incluyen la precisión del modelo (por ejemplo, mediante MAE, RMSE) y la utilidad práctica del pronóstico para la planificación de producción.
2. **Comprensión de los datos:** Se utilizan datos históricos de ventas y de inversión publicitaria. Los datos de ventas provienen de la API de Shopify e incluyen información como productos vendidos por día, canales de venta y comportamiento del usuario. La inversión publicitaria se obtiene desde la API de Triple Whale, con detalle de los montos invertidos en diferentes plataformas como Google, Facebook, Instagram y Klaviyo. El conjunto de datos abarca aproximadamente dos años y presenta una buena cobertura temporal, aunque puede haber días con registros faltantes o incompletos.
3. **Preparación de los datos:** Para el entrenamiento del modelo fue necesario consolidar los datos provenientes de Shopify y Triple Whale, generando un único dataset unificado por fecha. Se aplicaron transformaciones como limpieza de datos, imputación de valores faltantes, conversión de tipos y normalización de variables. Además, se generaron variables derivadas, como el promedio móvil de ventas o acumulados semanales de inversión.

4. **Modelado:** El problema a resolver es de tipo regresión, ya que se busca predecir una variable continua (ventas diarias). Se consideraron distintos algoritmos supervisados como regresión lineal, árbol de decisión, random forest, XGBoost y redes neuronales. La selección del modelo se basó en experimentación con validación cruzada.
5. **Evaluación del modelo:** La evaluación del rendimiento se realizó utilizando métricas como el MAE (Mean Absolute Error), RMSE (Root Mean Square Error) y MAPE (Mean Absolute Percentage Error). También se analizó la capacidad del modelo para captar la estacionalidad y tendencias generales. La comparación entre modelos se hizo sobre un conjunto de validación y se eligió el que mostró mejor desempeño en términos de precisión y generalización.
6. **Despliegue del modelo (opcional):** El modelo entrenado se integrará dentro del sistema del cliente denominado Inventory Tracker. El despliegue se hará en un entorno productivo con periodicidad de actualización semanal. Las predicciones se mostrarán en el sistema mediante visualizaciones y estarán disponibles para descarga en formato CSV. Además, se contempla un sistema de alertas para advertir sobre datos faltantes que puedan comprometer la precisión del modelo.

9. Desglose del trabajo en tareas

1. Planificación del proyecto (30 h)
 - 1.1. Definir alcance y cronograma (10 h)
 - 1.2. Presentación de plan de trabajo (8 h)
 - 1.3. Reuniones de seguimiento con el cliente y partes interesadas (12 h)
2. HU1 (36 h)
 - 2.1. Revisión de la documentación técnica de las APIs de Shopify (6 h)
 - 2.2. Pruebas de conexión, autenticación y extracción de datos de prueba (10 h)
 - 2.3. Generar script para descargar y almacenar resultados en archivos o base de datos (10 h)
 - 2.4. Comprobar la ejecución del script (10 h)
3. HU2 (36 h)
 - 3.1. Revisión de la documentación técnica de las APIs de Shopify y Triple Whale (6 h)
 - 3.2. Pruebas de conexión, autenticación y extracción de datos de prueba (10 h)
 - 3.3. Generar script para descargar y almacenar resultados en archivos o base de datos (10 h)
 - 3.4. Comprobar la ejecución del script (10 h)
4. HU3 (90 h)
 - 4.1. Limpieza y normalización de datos de Shopify (15 h)
 - 4.2. Limpieza y normalización de datos de Triple Whale (15 h)
 - 4.3. Diseño y desarrollo del pipeline de consolidación de datasets (20 h)
 - 4.4. Implementación de procesos de detección y manejo de outliers (10 h)
 - 4.5. Implementación de estrategias de imputación de datos faltantes (10 h)

- 4.6. Pruebas y validación del pipeline de datos (10 h)
- 4.7. Documentación técnica del pipeline de datos (10 h)
- 5. HU4 (160 h)
 - 5.1. Investigación y selección de algoritmos y técnicas de predicción (20 h)
 - 5.2. Implementación inicial de modelos candidatos (35 h)
 - 5.3. Desarrollo de scripts para evaluación y comparación de modelos (20 h)
 - 5.4. Optimización y ajuste de hiperparámetros (25 h)
 - 5.5. Validación cruzada y análisis de métricas de desempeño (20 h)
 - 5.6. Generación de documentación técnica detallada del modelo (20 h)
 - 5.7. Preparación de ejemplos de uso del modelo para integración (20 h)
- 6. HU5 (50 h)
 - 6.1. Diseñar el componente de visualización (20 h)
 - 6.2. Crear un endpoint que devuelva las predicciones de forma estructurada (10 h)
 - 6.3. Desarrollar el *frontend* para mostrar predicciones (20 h)
- 7. HU6 (50 h)
 - 7.1. Definir estructura del CSV (10 h)
 - 7.2. Implementar generación de CSV en *backend* (20 h)
 - 7.3. Desarrollar botón de descarga en frontend (10 h)
 - 7.4. Verificar que el archivo se descargue correctamente en distintos navegadores (10 h)
- 8. HU7 (80 h)
 - 8.1. Definir criterios de datos faltantes significativos (20 h)
 - 8.2. Desarrollar lógica de detección en *backend* (20 h)
 - 8.3. Generar alertas automáticas (20 h)
 - 8.4. Mostrar en el sistema si hay advertencias activas o todo está correcto (10 h)
 - 8.5. Testear escenarios con datos incompletos (10 h)
- 9. HU8 (152 h)
 - 9.1. Redacción del manual de usuario (15 h)
 - 9.2. Preparación de reportes técnicos y resultados de pruebas (15 h)
 - 9.3. Elaboración de diagramas de arquitectura y flujo de datos (10 h)
 - 9.4. Entregables de gestión de proyecto (30 h)
 - 9.5. Escritura de la memoria del trabajo versión 1 (30 h)
 - 9.6. Escritura de la memoria del trabajo de final (40 h)
 - 9.7. Preparación de presentaciones para exposición del proyecto (12 h)

Cantidad total de horas: 684 h

Historia de usuario	Tarea técnica	Estimación	Prioridad
Planificación	Tarea 1.1	10 h	Alta
Planificación	Tarea 1.2	8 h	Alta
Planificación	Tarea 1.3	12 h	Alta
HU1	Tarea 2.1 HU1	6 h	Baja
HU1	Tarea 2.2 HU1	10 h	Media
HU1	Tarea 2.3 HU1	10 h	Alta
HU1	Tarea 2.4 HU1	10 h	Alta
HU2	Tarea 3.1 HU2	6 h	Baja
HU2	Tarea 3.2 HU2	10 h	Media
HU2	Tarea 3.3 HU2	10 h	Alta
HU2	Tarea 3.4 HU2	10 h	Alta
HU3	Tarea 4.1 HU3	15 h	Alta
HU3	Tarea 4.2 HU3	15 h	Alta
HU3	Tarea 4.3 HU3	20 h	Media
HU3	Tarea 4.4 HU3	10 h	Media
HU3	Tarea 4.5 HU3	10 h	Media
HU3	Tarea 4.6 HU3	10 h	Media
HU3	Tarea 4.7 HU3	10 h	Media
HU4	Tarea 5.1 HU4	20 h	Alta
HU4	Tarea 5.2 HU4	35 h	Alta
HU4	Tarea 5.3 HU4	20 h	Alta
HU4	Tarea 5.4 HU4	25 h	Media
HU4	Tarea 5.5 HU4	20 h	Media
HU4	Tarea 5.6 HU4	20 h	Baja
HU4	Tarea 5.7 HU4	20 h	Baja
HU5	Tarea 6.1 HU5	20 h	Alta
HU5	Tarea 6.2 HU5	10 h	Media
HU5	Tarea 6.3 HU5	20 h	Alta
HU6	Tarea 7.1 HU6	10 h	Alta
HU6	Tarea 7.2 HU6	20 h	Alta
HU6	Tarea 7.3 HU6	10 h	Media
HU6	Tarea 7.4 HU6	10 h	Baja
HU7	Tarea 8.1 HU7	20 h	Alta
HU7	Tarea 8.2 HU7	20 h	Media
HU7	Tarea 8.3 HU7	20 h	Alta
HU7	Tarea 8.4 HU7	10 h	Media
HU7	Tarea 8.5 HU7	10 h	Baja
HU8	Tarea 9.1 HU8	15 h	Baja
HU8	Tarea 9.2 HU8	15 h	Media
HU8	Tarea 9.3 HU8	10 h	Media
HU8	Tarea 9.4 HU8	30 h	Alta
HU8	Tarea 9.5 HU8	30 h	Alta
HU8	Tarea 9.6 HU8	40 h	Alta
HU8	Tarea 9.7 HU8	12 h	Alta

6. Requerimientos

1. Requerimientos funcionales:

- 1.1. El sistema debe permitir importar datos históricos de ventas desde la *API* de Shopify. (Prioridad alta).
- 1.2. El sistema debe permitir importar datos históricos de inversión publicitaria desde la *API* de Triple Whale. (Prioridad alta).
- 1.3. El sistema debe consolidar y procesar los datos provenientes de ambas plataformas para crear un conjunto de datos único. (Prioridad alta).
- 1.4. El sistema debe entrenar modelos de inteligencia artificial capaces de predecir las ventas del producto por período de tiempo (día, semana, mes). (Prioridad alta).
- 1.5. El sistema debe permitir actualizar periódicamente las predicciones, mediante una ejecución programada. (Prioridad alta).
- 1.6. El usuario debe poder consultar las predicciones de ventas a través del sistema Inventory Tracker, integrado de forma nativa. (Prioridad alta).
- 1.7. El sistema debe permitir seleccionar el período de tiempo sobre el cual realizar las predicciones. (Prioridad media).
- 1.8. El sistema debe mantener la confidencialidad y seguridad de la información procesada. (Prioridad alta).

2. Requerimientos de documentación:

- 2.1. Documentación técnica que describa la arquitectura del sistema, el proceso de integración y el funcionamiento del modelo predictivo. (Prioridad alta).
- 2.2. Manual de usuario para la operación del módulo dentro de Inventory Tracker. (Prioridad media).
- 2.3. Documentación de instalación y configuración del módulo desarrollado. (Prioridad media).

3. Requerimiento de testing:

- 3.1. El sistema debe contar con pruebas unitarias sobre los componentes críticos. (Prioridad alta).
- 3.2. El sistema debe ser validado mediante pruebas funcionales integradas en Inventory Tracker. (Prioridad alta).
- 3.3. El modelo predictivo debe ser evaluado usando métricas estadísticas para asegurar una precisión mayor a 90 %. (Prioridad alta).

4. Requerimientos de la interfaz:

- 4.1. La interfaz de usuario dentro de Inventory Tracker debe permitir visualizar las predicciones de ventas de manera clara y comprensible. (Prioridad alta).
- 4.2. Las predicciones deben presentarse con opciones gráficas (por ejemplo, gráficos de series temporales) para facilitar su interpretación. (Prioridad media).
- 4.3. El sistema debe permitir la descarga de las predicciones y datos consolidados en formato *CSV* para su análisis externo. (Prioridad media).

5. Requerimientos de interoperabilidad:

- 5.1. El sistema debe integrarse sin problemas con la infraestructura existente de Inventory Tracker. (Prioridad alta).
- 5.2. El sistema debe adaptarse a posibles cambios en las *APIs* de Shopify y Triple Whale, siempre que estos cambios no impliquen modificaciones sustanciales en los datos entregados. (Prioridad media).
6. Requerimientos legales y regulatorios:
 - 6.1. El sistema debe cumplir con las leyes y regulaciones vigentes en materia de protección de datos personales. (Prioridad alta).
 - 6.2. La información procesada y almacenada no debe ser divulgada a terceros, en cumplimiento con el acuerdo de confidencialidad firmado con Latech. (Prioridad alta).

7. Historias de usuarios (*Product backlog*)

1. Épica 1: Integración de fuentes de datos
 - 1.1. “Como analista de datos quiero importar datos de ventas desde Shopify para poder entrenar el modelo de predicción.”
Story points: 5 (complejidad: 2, dificultad: 2, incertidumbre: 1)
 - 1.2. “Como analista de datos quiero importar datos de inversión publicitaria desde Triple Whale para incluir la variable de inversión en el análisis de ventas.”
Story points: 5 (complejidad: 2, dificultad: 2, incertidumbre: 1)
 - 1.3. “Como analista de datos quiero consolidar los datos de Shopify y Triple Whale para tener un único dataset listo para el entrenamiento del modelo.”
Story points: 8 (complejidad: 3, dificultad: 2, incertidumbre: 3)
2. Épica 2: Modelado predictivo
 - 2.1. “Como analista de datos quiero entrenar un modelo que prediga las ventas para poder anticipar la demanda de producción.”
Story points: 13 (complejidad: 5, dificultad: 4, incertidumbre: 4)
3. Épica 3: Visualización y exportación de resultados
 - 3.1. “Como usuario de Inventory Tracker quiero visualizar las predicciones de ventas dentro del sistema para poder planificar la producción.”
Story points: 8 (complejidad: 3, dificultad: 2, incertidumbre: 3)
 - 3.2. “Como usuario de Inventory Tracker quiero descargar las predicciones en formato CSV para analizarlas externamente.”
Story points: 3 (complejidad: 1, dificultad: 1, incertidumbre: 1)
4. Épica 4: Monitoreo y alertas del sistema
 - 4.1. “Como responsable del sistema quiero recibir alertas si hay datos faltantes que puedan afectar significativamente la precisión de las predicciones.”
Story points: 5 (complejidad: 2, dificultad: 2, incertidumbre: 1)

8. Entregables principales del proyecto

- Manual de usuario.
- Diagramas de arquitectura del sistema y de integración con Inventory Tracker.
- Código fuente del software para procesamiento de datos, entrenamiento de modelos y generación de predicciones.
- Diagrama de flujo del proceso de carga, procesamiento y predicción de datos.
- Memoria del trabajo final.
- Documentación técnica detallada sobre la instalación, configuración y mantenimiento del módulo desarrollado.

9. Desglose del trabajo en tareas

1. Planificación del proyecto (30 h)
 - 1.1. Definir alcance y cronograma (10 h)
 - 1.2. Presentación de plan de trabajo (8 h)
 - 1.3. Reuniones de seguimiento con el cliente y partes interesadas (12 h)
2. Análisis y relevamiento de datos (72 h)
 - 2.1. Revisión de la documentación técnica de las APIs de Shopify y Triple Whale (12 h)
 - 2.2. Pruebas de conexión, autenticación y extracción de datos de prueba (20 h)
 - 2.3. Análisis exploratorio de datos (EDA) sobre datasets extraídos (20 h)
 - 2.4. Identificación de variables relevantes para el modelo (10 h)
 - 2.5. Documentación de hallazgos del análisis de datos (10 h)
3. Preparación y procesamiento de datos (90 h)
 - 3.1. Limpieza y normalización de datos de Shopify (15 h)
 - 3.2. Limpieza y normalización de datos de Triple Whale (15 h)
 - 3.3. Diseño y desarrollo del pipeline de consolidación de datasets (20 h)
 - 3.4. Implementación de procesos de detección y manejo de outliers (10 h)
 - 3.5. Implementación de estrategias de imputación de datos faltantes (10 h)
 - 3.6. Pruebas y validación del pipeline de datos (10 h)
 - 3.7. Documentación técnica del pipeline de datos (10 h)
4. Desarrollo del modelo predictivo (120 h)
 - 4.1. Investigación y selección de algoritmos y técnicas de predicción (15 h)
 - 4.2. Implementación inicial de modelos candidatos (30 h)
 - 4.3. Desarrollo de scripts para evaluación y comparación de modelos (15 h)
 - 4.4. Optimización y ajuste de hiperparámetros (20 h)
 - 4.5. Validación cruzada y análisis de métricas de desempeño (20 h)

- 4.6. Generación de documentación técnica detallada del modelo (10 h)
- 4.7. Preparación de ejemplos de uso del modelo para integración (10 h)
- 5. Desarrollo del módulo e integración (140 h)
 - 5.1. Diseño de la arquitectura del módulo integrado en Inventory Tracker (15 h)
 - 5.2. Desarrollo backend para ejecución del modelo y manejo de resultados (30 h)
 - 5.3. Desarrollo de API interna para comunicación entre el módulo y el resto del sistema (20 h)
 - 5.4. Diseño y desarrollo de la interfaz de usuario para visualización de resultados (20 h)
 - 5.5. Implementación de descarga de predicciones en formato CSV (10 h)
 - 5.6. Implementación de mecanismos de alerta por datos inconsistentes o incompletos (10 h)
 - 5.7. Pruebas de integración en entorno de desarrollo (15 h)
 - 5.8. Documentación técnica de integración (10 h)
 - 5.9. Soporte técnico para puesta en marcha en ambiente productivo (10 h)
- 6. Prueba y validación (80 h)
 - 6.1. Diseño de plan de pruebas funcionales y técnicas (10 h)
 - 6.2. Desarrollo de pruebas unitarias (15 h)
 - 6.3. Ejecución de pruebas funcionales del módulo completo (15 h)
 - 6.4. Validación de resultados de predicciones con datos históricos (15 h)
 - 6.5. Análisis de casos extremos y situaciones de error (10 h)
 - 6.6. Generación de reportes de resultados de pruebas (15 h)
- 7. Documentación y entregables (152 h)
 - 7.1. Redacción del manual de usuario (15 h)
 - 7.2. Preparación de reportes técnicos y resultados de pruebas (15 h)
 - 7.3. Elaboración de diagramas de arquitectura y flujo de datos (10 h)
 - 7.4. Entregables de gestión de proyecto (30 h)
 - 7.5. Escritura de la memoria del trabajo versión 1 (30 h)
 - 7.6. Escritura de la memoria del trabajo de final (40 h)
 - 7.7. Preparación de presentaciones para exposición del proyecto (12 h)

Cantidad total de horas: 684 h

10. Planificación de Sprints

Sprint	HU o fase	Tarea	Horas / SP	Responsable	% Completado
0	Planificación	Definir alcance y cronograma	10 h / 5 SP	Alumno	80 %
0	Planificación	Presentación de plan de trabajo	8 h / 4 SP	Alumno	0 %
0	Planificación	Reuniones de seguimiento iniciales	12 h / 6 SP	Alumno	0 %
1	HU1	Revisión documentación APIs Shopify/Triple Whale	12 h / 6 SP	Alumno	0 %
1	HU2	Pruebas conexión y extracción datos prueba	20 h / 10 SP	Alumno	0 %
2	HU3	Análisis exploratorio de datos (EDA)	20 h / 10 SP	Alumno	0 %
2	HU3	Identificación variables relevantes	10 h / 5 SP	Alumno	0 %
3	HU4	Documentación hallazgos EDA	10 h / 5 SP	Alumno	0 %
3	HU4	Limpieza y normalización datos Shopify	15 h / 8 SP	Alumno	0 %
3	HU4	Limpieza y normalización datos Triple Whale	15 h / 8 SP	Alumno	0 %
4	HU4	Diseño y desarrollo pipeline datasets	20 h / 10 SP	Alumno	0 %
4	HU4	Procesos detección/manejo outliers	10 h / 5 SP	Alumno	0 %
5	HU4	Imputación datos faltantes	10 h / 5 SP	Alumno	0 %
5	HU4	Pruebas y validación pipeline	10 h / 5 SP	Alumno	0 %
5	HU4	Documentación técnica pipeline	10 h / 5 SP	Alumno	0 %
6	HU4	Investigación algoritmos predicción	15 h / 8 SP	Alumno	0 %
6	HU4	Implementación inicial modelos candidatos	30 h / 15 SP	Alumno	0 %

Sprint	HU o fase	Tarea	Horas / SP	Responsable	% Completado
7	HU4	Scripts evaluación y comparación modelos	15 h / 8 SP	Alumno	0 %
7	HU4	Optimización hiperparámetros	20 h / 10 SP	Alumno	0 %
8	HU4	Validación cruzada y métricas desempeño	20 h / 10 SP	Alumno	0 %
8	HU4	Documentación técnica modelo	10 h / 5 SP	Alumno	0 %
8	HU4	Ejemplos de uso del modelo	10 h / 5 SP	Alumno	0 %
9	HU5	Arquitectura módulo Inventory Tracker	15 h / 8 SP	Alumno	0 %
9	HU5	Desarrollo backend ejecución modelo	30 h / 15 SP	Alumno	0 %
10	HU5	Desarrollo API interna	20 h / 10 SP	Alumno	0 %
10	HU5	Interfaz visualización resultados	20 h / 10 SP	Alumno	0 %
11	HU6	Descarga predicciones CSV	10 h / 5 SP	Alumno	0 %
11	HU7	Mecanismos alerta datos inconsistentes	10 h / 5 SP	Alumno	0 %
11	HU7	Pruebas integración desarrollo	15 h / 8 SP	Alumno	0 %
12	Documentación	Documentación técnica integración	10 h / 5 SP	Alumno	0 %
12	Despliegue	Soporte puesta en marcha producción	10 h / 5 SP	Alumno	0 %
12	Pruebas	Diseño plan pruebas	10 h / 5 SP	Alumno	0 %
12	Pruebas	Desarrollo pruebas unitarias	15 h / 8 SP	Alumno	0 %
12	Pruebas	Ejecución pruebas funcionales	15 h / 8 SP	Alumno	0 %
12	Pruebas	Validación predicciones históricos	15 h / 8 SP	Alumno	0 %

Sprint	HU o fase	Tarea	Horas / SP	Responsable	% Completado
12	Pruebas	Casos extremos y errores	10 h / 5 SP	Alumno	0 %
12	Pruebas	Reportes pruebas	15 h / 8 SP	Alumno	0 %
13	Documentación	Manual de usuario	15 h / 8 SP	Alumno	0 %
13	Documentación	Reportes técnicos y resultados	15 h / 8 SP	Alumno	0 %
13	Documentación	Diagramas arquitectura y flujo datos	10 h / 5 SP	Alumno	0 %
13	Documentación	Entregables gestión proyecto	30 h / 15 SP	Alumno	0 %
14	Documentación	Escritura memoria versión 1	30 h / 15 SP	Alumno	0 %
14	Documentación	Escritura memoria final	40 h / 20 SP	Alumno	0 %
14	Documentación	Preparación exposición proyecto	12 h / 6 SP	Alumno	0 %

11. Diagrama de Gantt (sprints)

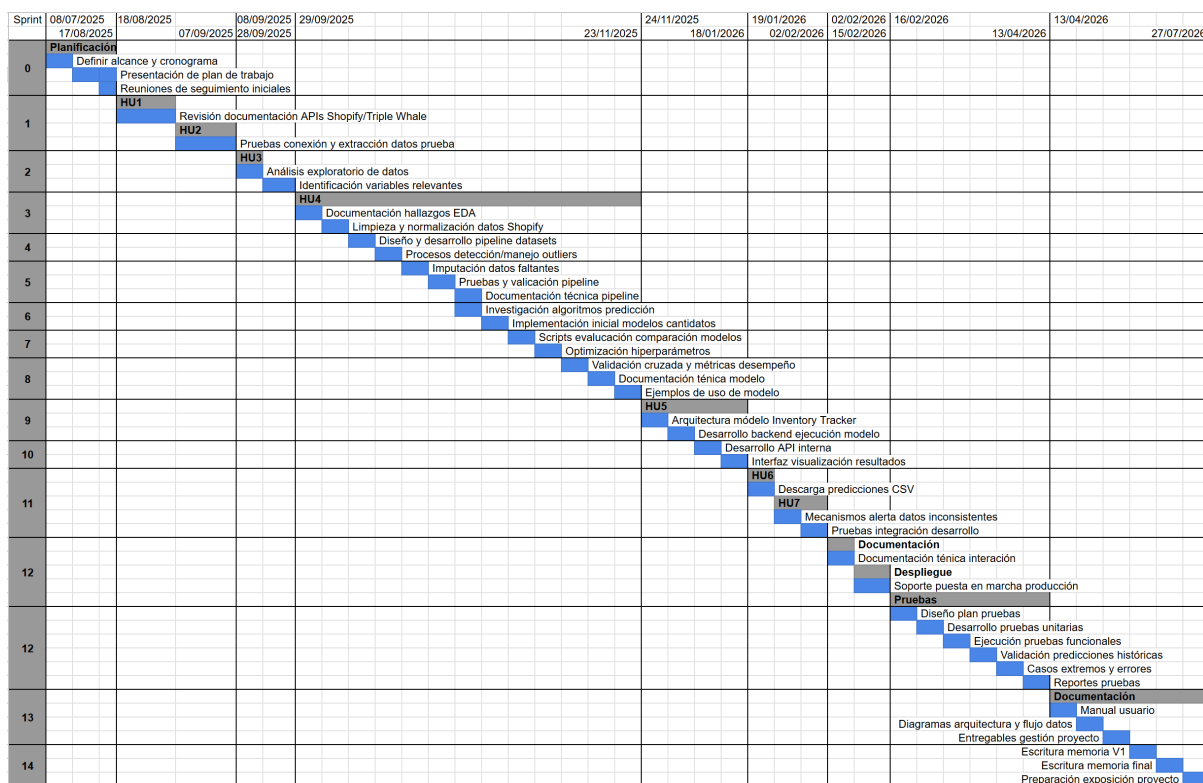


Figura 2. Diagrama Gantt.

12. Normativa y cumplimiento de datos (gobernanza)

Los datos que se utilizarán en este proyecto contendrán identificadores únicos de usuarios obtenidos mediante una API privada de Shopify autorizada por la empresa cliente. El acceso será provisto exclusivamente con fines académicos y de desarrollo, y se realizará de acuerdo con los Términos de Servicio de Shopify.

Se tomarán medidas de ofuscación para evitar la exposición de información sensible, eliminando o codificando los datos identificables antes de su análisis o documentación. Asimismo, se aplicará el principio de minimización de datos, solicitando únicamente la información estrictamente necesaria para el desarrollo del modelo de predicción.

El uso de estos datos estará restringido por la Ley 25.326 de Protección de Datos Personales en Argentina y se prohibirá su difusión, redistribución o reutilización con otros fines. Además, se garantizarán condiciones de confidencialidad, evitando su publicación en reportes, repositorios o cualquier medio accesible a terceros no autorizados.

Estas acciones permitirán asegurar el cumplimiento normativo, la ética en el tratamiento de los datos y la protección de los derechos de los usuarios involucrados.

13. Gestión de riesgos

Riesgo 1: Baja calidad o inconsistencia de los datos obtenidos desde las APIs

- Severidad (S): 8 Una baja calidad en los datos afectaría directamente la precisión del modelo de predicción, comprometiendo la utilidad del sistema para la toma de decisiones.
- Ocurrencia (O): 6 Aunque las *APIs* de Shopify y Triple Whale son robustas, pueden presentarse inconsistencias o errores de carga en ciertos períodos, sobre todo en integraciones recientes.

Riesgo 2: Cambios en las *APIs* externas

- Severidad (S): 7 Si se modifican los endpoints o se restringe el acceso a ciertas métricas, se vería afectada la actualización de datos históricos y la operación del sistema.
- Ocurrencia (O): 5 Este tipo de cambios no son frecuentes, pero pueden ocurrir sin previo aviso por decisión unilateral de los proveedores.

Riesgo 3: Sobreajuste del modelo (*overfitting*)

- Severidad (S): 6 Si el modelo se ajusta demasiado a los datos históricos, podría perder capacidad de generalización, resultando en malas predicciones a futuro.
- Ocurrencia (O): 5 Es un riesgo habitual en proyectos de *machine learning*, aunque puede mitigarse con técnicas adecuadas de validación cruzada.

Riesgo 4: Incumplimiento del cronograma por complejidad técnica

- Severidad (S): 5 Un retraso significativo afectaría los tiempos de entrega al cliente y la integración con otros sistemas.
- Ocurrencia (O): 7 Dada la complejidad de integrar varias fuentes y construir un modelo robusto, es un riesgo relevante.

Riesgo 5: Filtración o uso indebido de datos sensibles

- Severidad (S): 9 La exposición de datos identificables como correos electrónicos podría implicar consecuencias legales o de reputación graves.
- Ocurrencia (O): 3 El riesgo es bajo ya que se prevé la implementación de medidas de ofuscación y restricciones de acceso.

Riesgo 6: Pérdida del vínculo laboral con la empresa cliente durante el desarrollo del proyecto

Justificación: Existe la posibilidad de que durante el desarrollo del trabajo final se disuelva el vínculo laboral actual con la empresa Latech. En ese caso, se podría dificultar el acceso a datos, herramientas o validaciones necesarias para completar el proyecto, afectando su continuidad.

- Severidad (S): 9 La pérdida del acceso a recursos o información crítica podría implicar la necesidad de replantear partes sustanciales del proyecto o incluso buscar un nuevo cliente.

- Ocurrencia (O): 4 Aunque actualmente el vínculo es estable, no puede descartarse completamente un cambio de situación laboral.

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

Criterio adoptado: Se tomarán medidas de mitigación para aquellos riesgos cuyo RPN supere los 30 puntos.

Riesgo	S	O	RPN	S*	O*	RPN*
Baja calidad de datos	8	6	48	6	4	24
Cambios en las APIs	7	5	35	6	3	18
Sobreajuste del modelo	6	5	30	4	3	12
Demoras en cronograma	5	7	35	3	4	12
Filtración de datos	9	3	27	4	2	8
Desvinculación laboral	9	4	36	4	2	8

c) Plan de mitigación de riesgos

Riesgo 1: Baja calidad de datos

Plan de mitigación: Se establecerán rutinas automáticas de validación y limpieza de datos antes de ser procesados por el modelo.

- Severidad (S): 6* La calidad seguirá siendo importante, pero con validación previa se reduce su impacto.
- Ocurrencia (O): 4* La verificación regular reducirá las inconsistencias.

Riesgo 2: Cambios en las APIs

Plan de mitigación: Se desarrollarán capas intermedias (*wrappers*) para desacoplar el sistema de los cambios en las APIs. Además, se monitorearán las actualizaciones oficiales.

- Severidad (S): 6* Los cambios seguirán siendo relevantes, pero se reducirá su impacto directo.
- Ocurrencia (O): 3* La supervisión reduce la probabilidad de fallos inesperados.

Riesgo 4: Demoras en el cronograma

Plan de mitigación: Se trabajará en ciclos iterativos con entregas parciales, priorizando funcionalidades clave para asegurar un producto mínimo viable en tiempo.

- Severidad (S): 3* Las demoras parciales no afectarán el objetivo final.
- Ocurrencia (O): 4* La planificación iterativa mejora el control del avance.

Riesgo 6: Desvinculación laboral

Plan de mitigación:

Se acordará con la empresa la continuidad del acceso a los datos y a las herramientas necesarias para completar el trabajo final, aún en el caso de que se termine la relación laboral. Esto podría contemplarse en un documento formal o compromiso verbal con respaldo escrito.

- Severidad (S): 4* Si se garantiza el acceso a los recursos clave, el impacto se reduce notablemente.
- Ocurrencia (O): 2* Al prever esta situación y dejarla documentada, disminuye la probabilidad de que se interrumpa el acceso al entorno de trabajo.

a) Identificación de los riesgos (al menos cinco) y estimación de sus consecuencias:

Riesgo 1: detallar el riesgo (riesgo es algo que si ocurre altera los planes previstos de forma negativa)

- Severidad (S): mientras más severo, más alto es el número (usar números del 1 al 10). Justificar el motivo por el cual se asigna determinado número de severidad (S).
- Probabilidad de ocurrencia (O): mientras más probable, más alto es el número (usar del 1 al 10). Justificar el motivo por el cual se asigna determinado número de (O).

Riesgo 2:

- Severidad (S): X.
Justificación...
- Ocurrencia (O): Y.
Justificación...

Riesgo 3:

- Severidad (S): X.
Justificación...
- Ocurrencia (O): Y.
Justificación...

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

Riesgo	S	O	RPN	S*	O*	RPN*

Criterio adoptado:

Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a...

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: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación). Nueva asignación de S y O, con su respectiva justificación:

- Severidad (S*): mientras más severo, más alto es el número (usar números del 1 al 10). Justificar el motivo por el cual se asigna determinado número de severidad (S).
- Probabilidad de ocurrencia (O*): mientras más probable, más alto es el número (usar del 1 al 10). Justificar el motivo por el cual se asigna determinado número de (O).

Riesgo 2: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación).

Riesgo 3: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación).

14. Sprint Review

La revisión de sprint (*Sprint Review*) es una práctica fundamental en metodologías ágiles. Consiste en revisar y evaluar lo que se ha completado al finalizar un sprint. En esta instancia, se presentan los avances y se verifica si las funcionalidades cumplen con los criterios de aceptación establecidos. También se identifican entregables parciales y se consideran ajustes si es necesario.

Aunque el proyecto aún se encuentre en etapa de planificación, esta sección permite proyectar cómo se evaluarán las funcionalidades más importantes del backlog. Esta mirada anticipada favorece la planificación enfocada en valor y permite reflexionar sobre posibles obstáculos.

Objetivo: anticipar cómo se evaluará el avance del proyecto a medida que se desarrollen las funcionalidades, utilizando como base al menos cuatro historias de usuario del *Product Backlog*.

Seleccionar al menos 4 HU del Product Backlog. Para cada una, completar la siguiente tabla de revisión proyectada:

Formato sugerido:

HU seleccionada	Tareas asociadas	Entregable esperado	¿Cómo sabrás que está cumplida?	Observaciones o riesgos
HU1	Tarea 1	Módulo funcional	Cumple criterios de aceptación definidos	Falta validar con el tutor
	Tarea 2			
HU3	Tarea 1	Reporte generado	Exportación disponible y clara	Requiere datos reales
	Tarea 2			
HU5	Tarea 1	Panel de gestión	Roles diferenciados operativos	Riesgo en integración
	Tarea 2			
HU7	Tarea 1	Informe trimestral	PDF con gráficos y evolución	Puede faltar tiempo para ajustes
	Tarea 2			

15. Sprint Retrospective

La retrospectiva de sprint es una práctica orientada a la mejora continua. Al finalizar un sprint, el equipo (o el alumno, si trabaja de forma individual) reflexiona sobre lo que funcionó bien, lo que puede mejorarse y qué acciones concretas pueden implementarse para trabajar mejor en el futuro.

Durante la cursada se propuso el uso de la **Estrella de la Retrospectiva**, que organiza la reflexión en torno a cinco ejes:

- ¿Qué hacer más?
- ¿Qué hacer menos?
- ¿Qué mantener?
- ¿Qué empezar a hacer?
- ¿Qué dejar de hacer?

Aun en una etapa temprana, esta herramienta permite que el alumno planifique su forma de trabajar, identifique anticipadamente posibles dificultades y diseñe estrategias de organización personal.

Objetivo: reflexionar sobre las condiciones iniciales del proyecto, identificando fortalezas, posibles dificultades y estrategias de mejora, incluso antes del inicio del desarrollo.

Completar la siguiente tabla tomando como referencia los cinco ejes de la Estrella de la Retrospectiva (*Starfish* o estrella de mar). Esta instancia te ayudará a definir buenas prácticas desde el inicio y prepararte para enfrentar el trabajo de forma organizada y flexible. Se deberá completar la tabla al menos para 3 sprints técnicos y 1 no técnico.

Formato sugerido:

Sprint tipo y N°	¿Qué hacer más?	¿Qué hacer menos?	¿Qué mantener?	¿Qué empezar a hacer?	¿Qué dejar de hacer?
Sprint técnico - 1	Validaciones continuas con el alumno	Cambios sin versión registrada	Pruebas con datos simulados	Documentar cambios propuestos	Ajustes sin análisis de impacto
Sprint técnico - 2	Verificar configuraciones en múltiples escenarios	Modificar parámetros sin guardar historial	Perfiles reutilizables	Usar logs para configuración	Repetir pruebas manuales innecesarias
Sprint técnico - 8	Comparar correlaciones con casos previos	Cambiar parámetros sin justificar	Revisión cruzada de métricas	Anotar configuraciones usadas	Trabajar sin respaldo de datos
Sprint no técnico - 12 (por ej.: “Defensa”)	Ensayos orales con feedback	Cambiar contenidos en la memoria	Material visual claro	Dividir la presentación por bloques	Agregar gráficos difíciles de explicar