



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. Requerimientos	8
7. Historias de usuarios (<i>Product backlog</i>).	9
8. Entregables principales del proyecto	10
9. Desglose del trabajo en tareas	10
10. Planificación de Sprints	12
11. Diagrama de Gantt (sprints)	15
12. Normativa y cumplimiento de datos (gobernanza)	16
13. Gestión de riesgos	16
14. Sprint Review	17
15. Sprint Retrospective	18

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 \$ XXX, 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.

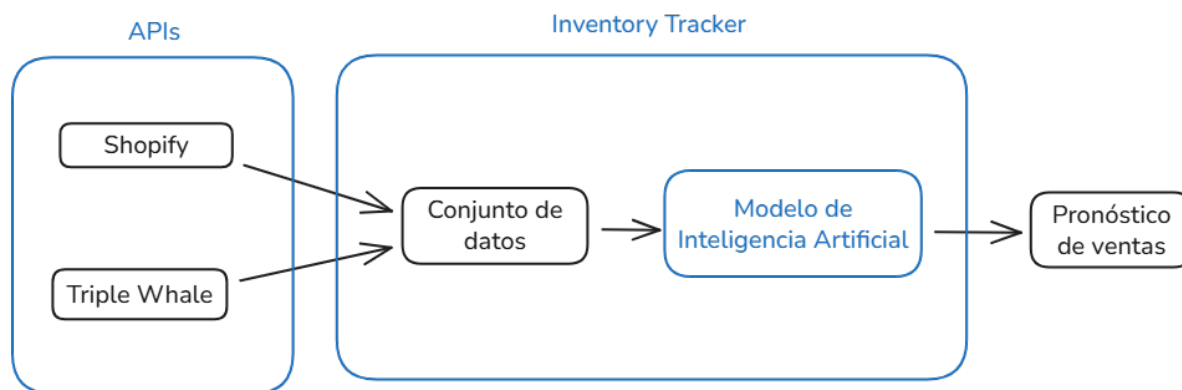


Figura 1. Diagrama en bloques del sistema.

2. Identificación y análisis de los interesados

Cuadro 1. Identificación 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. 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

Organizar las tareas técnicas del proyecto en sprints de trabajo que permitan distribuir de forma equilibrada la carga horaria total, estimada en 600 horas.

Consigna:

- Completar una tabla que relacione sprints con HU y tareas técnicas correspondientes.
- Incluir estimación en horas para cada tarea.
- Indicar responsable y porcentaje de avance estimado o completado.
- Contemplar también tareas de planificación, documentación, redacción de memoria y preparación de defensa.

Conceptos clave:

- Una épica es una unidad funcional amplia; una historia de usuario es una funcionalidad concreta; un sprint es una unidad de tiempo donde se ejecutan tareas.
- Las tareas son el nivel más desagregado: permiten estimar tiempos, asignar responsables y monitorear progreso.

Duración sugerida:

- Para un proyecto de 600 h, se recomienda planificar entre 10 y 12 sprints de aproximadamente 2 semanas cada uno.
- Asignar entre 45 y 50 horas efectivas por sprint a tareas técnicas.
- Reservar 100 a 120 h para actividades no técnicas (planificación, escritura, reuniones, defensa).

Importante:

- En proyectos individuales, el responsable suele ser el propio autor.
- Aun así, desagregar tareas facilita el seguimiento y mejora continua.

Conversión opcional de Story Points a horas:

- $1 \text{ SP} \approx 2 \text{ h}$ como referencia flexible.
- Tener en cuenta aproximaciones tipo Fibonacci.

Recomendaciones:

- Verificar que la carga horaria por sprint sea equilibrada.
- Usar sprints de 1 a 3 semanas, acordes al cronograma general.
- Actualizar el % completado durante el seguimiento del proyecto.
- Considerar un sprint final exclusivo para pruebas, revisión y ajustes antes de la defensa.

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 %
12	Documentación	Manual de usuario	15 h / 8 SP	Alumno	0 %
12	Documentación	Reportes técnicos y resultados	15 h / 8 SP	Alumno	0 %
12	Documentación	Diagramas arquitectura y flujo datos	10 h / 5 SP	Alumno	0 %
12	Documentación	Entregables gestión proyecto	30 h / 15 SP	Alumno	0 %
12	Documentación	Escritura memoria versión 1	30 h / 15 SP	Alumno	0 %
12	Documentación	Escritura memoria final	40 h / 20 SP	Alumno	0 %
12	Documentación	Preparación exposición proyecto	12 h / 6 SP	Alumno	0 %

11. Diagrama de Gantt (sprints)

Visualizar en un diagrama de Gantt la planificación temporal del proyecto, tomando como base los sprints definidos en la sección anterior. Debe contemplar todas las horas del proyecto.

Consigna:

- Elaborar un diagrama de Gantt que muestre la secuencia temporal de los sprints.
- Cada fila debe representar un sprint (con su número o nombre), y el eje horizontal debe indicar el tiempo (en semanas o fechas concretas).
- Las tareas técnicas derivadas de HU deben diferenciarse visualmente (por ejemplo, con un color distinto) de las tareas no técnicas (planificación, redacción, defensa).
- Incluir todas las tareas estimadas en cada sprint.

Recomendaciones para el Gantt:

- Podés usar herramientas gratuitas como TeamGantt, ClickUp, GanttProject, [Google Sheets], [Trello + Planyway], entre otras.
- Ordená los sprints de forma cronológica, comenzando con Sprint 0 (planificación) y finalizando con el sprint de defensa.
- Asegurate de reflejar la duración realista de cada sprint según tu disponibilidad y el cronograma general del posgrado.

- Incluí hitos importantes: reuniones, entregas parciales, defensa.

Incluir una imagen legible del diagrama de Gantt. Si es muy ancho, presentar primero la tabla y luego el gráfico de barras.

12. Normativa y cumplimiento de datos (gobernanza)

En esta sección se debe analizar si los datos utilizados en el proyecto están sujetos a normativas de protección de datos y privacidad, y en qué condiciones se pueden emplear.

Aspectos a considerar:

- Evaluar si los datos están regulados por normativas como GDPR, Ley 25.326 de Protección de Datos Personales en Argentina, HIPAA u otras según jurisdicción y temática.
- Determinar si el uso de los datos requiere consentimiento explícito de los usuarios involucrados.
- Indicar si existen restricciones legales, técnicas o contractuales sobre el uso, compartición o publicación de los datos.
- Aclarar si los datos provienen de fuentes licenciadas, de acceso público o bajo algún tipo de autorización especial.
- Analizar la viabilidad del proyecto desde el punto de vista legal y ético, considerando la gobernanza de los datos.

Este análisis es clave para garantizar el cumplimiento normativo y evitar conflictos legales durante el desarrollo y publicación del proyecto.

13. Gestión de riesgos

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...

- Ocurrecia (O): Y.
Justificación...

Riesgo 3:

- Severidad (S): X.
Justificación...
- Ocurrecia (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