## **Capítulo I: Definición del Proyecto**

### ****1.1 Origen del Proyecto****

El presente proyecto surge a partir de la necesidad de **democratizar el acceso a tecnologías de inteligencia artificial (IA)** aplicadas al análisis de ventas y la gestión de inventario en el sector minorista.  
Durante la etapa de relevamiento inicial se identificó una marcada brecha tecnológica entre las **grandes corporaciones**, que utilizan sistemas predictivos avanzados como SAP Integrated Business Planning o Oracle Retail, y las **pequeñas y medianas empresas (PyMEs)**, que continúan gestionando sus operaciones con métodos manuales o sistemas rígidos sin capacidad analítica.  
Este escenario evidenció una oportunidad concreta para desarrollar una **plataforma inteligente accesible y escalable**, capaz de ofrecer funciones de predicción de demanda y personalización de ofertas sin depender de infraestructuras costosas ni entornos cerrados.  
El proyecto combina enfoques de Machine Learning (ML) y Deep Learning, en particular redes neuronales LSTM, bajo un esquema metodológico de **MLOps** y **Spec-Driven Development**, que garantizan la trazabilidad, reproducibilidad y mejora continua durante el ciclo de vida del modelo.

### ****1.2 Misión, Visión y Objetivos****

**Misión:**  
Desarrollar una plataforma inteligente que permita a los comercios anticipar la demanda, optimizar el inventario y generar ofertas personalizadas para sus clientes mediante el uso de inteligencia artificial accesible, escalable y reproducible.

**Visión:**  
Consolidar una solución tecnológica integral que potencie la competitividad de las PyMEs a través del uso ético y eficiente de la IA, contribuyendo a la transformación digital del sector retail y fomentando la adopción de prácticas de análisis predictivo en entornos de bajo costo.

**Objetivo General:**  
Diseñar e implementar un **asistente inteligente de ventas y gestión de inventario** basado en modelos LSTM y metodologías MLOps, que prediga la demanda de productos, detecte tendencias de consumo y proponga estrategias de promoción personalizadas.

**Objetivos Específicos:**

1. Investigar el estado del arte sobre predicción de demanda en retail mediante técnicas de aprendizaje profundo.
2. Implementar un modelo LSTM univariado y multivariado que permita anticipar la demanda a corto y mediano plazo.
3. Integrar la gestión de experimentos con **MLflow** y automatizar el ciclo de vida del modelo bajo buenas prácticas de **MLOps**.
4. Desarrollar un módulo de recomendaciones personalizadas basado en el historial de compras de cada cliente.
5. Diseñar un **dashboard interactivo** que presente métricas, predicciones y alertas de stock en tiempo real.
6. Evaluar la escalabilidad del sistema en distintos escenarios de negocio, desde comercios locales hasta cadenas minoristas.

### ****1.3 Necesidad o Problema que Responde****

En el contexto actual de digitalización, las empresas dependen cada vez más del análisis de datos para la toma de decisiones estratégicas. Sin embargo, las **PyMEs carecen de herramientas predictivas asequibles** que les permitan optimizar su inventario o personalizar sus estrategias de venta.  
Esto genera múltiples problemáticas:

* **Quiebres de stock** por falta de previsión en la demanda.
* **Exceso de inventario inmovilizado**, que afecta la liquidez del negocio.
* **Promociones genéricas** sin un enfoque en el comportamiento real del cliente.

El sistema propuesto aborda estas limitaciones mediante una **plataforma inteligente de predicción de demanda y gestión comercial**, capaz de:

* Anticipar ventas futuras utilizando redes neuronales LSTM entrenadas con datos históricos.
* Ajustar el stock y generar alertas de reposición automática.
* Recomendar ofertas o combos personalizados por perfil de cliente, basándose en patrones de consumo.

De esta manera, el proyecto **transforma los datos de ventas en decisiones accionables**, reduciendo costos operativos y aumentando la eficiencia del negocio.

### ****1.4 ODS Asociadas y Diferenciales****

El proyecto se alinea con los **Objetivos de Desarrollo Sostenible (ODS)** propuestos por la ONU, principalmente con:

* **ODS 8: Trabajo decente y crecimiento económico**, al fortalecer la competitividad de las PyMEs mediante la incorporación de tecnología avanzada.
* **ODS 9: Industria, innovación e infraestructura**, promoviendo la digitalización del sector minorista a través de soluciones escalables e innovadoras.
* **ODS 12: Producción y consumo responsables**, al optimizar el manejo de inventarios y reducir desperdicios de productos por sobreoferta o caducidad.

**Diferenciales del Proyecto:**

* Aplicación de **inteligencia artificial accesible y modular**, adaptable a distintos tamaños de negocio.
* Integración de prácticas de **MLOps y Spec-Driven Development**, garantizando reproducibilidad y mantenibilidad.
* Capacidad de generar **recomendaciones personalizadas**, mejorando la experiencia de cliente mediante IA explicativa.
* Independencia de plataformas cerradas (SAP, Oracle, Amazon), permitiendo la **autonomía tecnológica** del usuario final.

### ****1.5 Descripción Breve del Sistema de Información****

El **Asistente Inteligente de Ventas y Gestión de Inventario** es una plataforma digital que emplea algoritmos de aprendizaje profundo para analizar los patrones de venta de productos, estimar la demanda futura y proponer estrategias de comercialización personalizadas.  
El sistema toma como insumo datasets de ventas (propios o de fuentes abiertas), los preprocesa y los transforma en información útil para la **toma de decisiones estratégicas** dentro del comercio.  
Su estructura modular permite operar tanto en entornos locales como en la nube, integrándose fácilmente con otros sistemas existentes de gestión (ERP o POS).

### ****1.6 Descripción Detallada del Sistema de Información****

El sistema propuesto se compone de tres capas funcionales principales:

1. **Capa de Datos:**  
   Encargada de la recolección, limpieza y almacenamiento de datos históricos de ventas. Integra datasets estructurados (por ejemplo, Walmart Product Demand Dataset de Kaggle) y registros internos del comercio.
2. **Capa de Modelado Predictivo:**  
   Implementa modelos basados en redes neuronales LSTM (Long Short-Term Memory) que aprenden patrones de comportamiento de los consumidores.  
   Esta capa incluye procesos automáticos de entrenamiento, validación y registro de experimentos mediante MLflow.  
   Los resultados se evalúan con métricas estándar como **MAE** y **RMSE**, garantizando objetividad en la comparación de modelos.
3. **Capa de Aplicación:**  
   Se encarga de la interacción con el usuario final a través de un **dashboard visual** que muestra predicciones, alertas de stock y sugerencias de ofertas personalizadas.  
   También se prevé la integración de un sistema de notificaciones automatizadas y reportes periódicos, configurables según las necesidades del comercio.

De manera transversal, el sistema aplica principios de **MLOps**, asegurando la trazabilidad, automatización y escalabilidad del ciclo de vida de los modelos, y de **Spec-Driven Development**, que define de forma contractual cada funcionalidad antes de su implementación.

## **Capítulo II: Relevamiento e Investigación de Mercado**

### ****2.1 Fuentes de Datos Utilizadas****

El relevamiento de información se llevó a cabo mediante un **Mapeo Sistemático de la Literatura (MSL)** enfocado en la aplicación de **redes neuronales LSTM (Long Short-Term Memory)** y modelos híbridos (CNN-LSTM, BiLSTM, GRU) a la predicción de demanda en el sector retail.  
Para garantizar rigor y reproducibilidad, las búsquedas se realizaron en bases académicas reconocidas —**IEEE Xplore, ACM Digital Library, ScienceDirect y Google Scholar**— y en repositorios de datos abiertos y proyectos experimentales como **Kaggle** y **GitHub**.1

Entre las fuentes seleccionadas se encuentran tres artículos de referencia central:

* [1] Predictive Analytics for Demand Forecasting – A Comparison of SARIMA and LSTM in Retail SCM (ScienceDirect).
* [2] Enhancing Time Series Product Demand Forecasting with Hybrid Attention-Based Deep Learning Models (IEEE).
* [3] A Hybrid Deep Learning Framework with CNN and Bi-LSTM for Store Item Demand Forecasting (ScienceDirect).

Además, se incorporaron datasets públicos relevantes para el entrenamiento de modelos predictivos, tales como:

* Product Prices and Sizes from Walmart Grocery [8],
* Warehouse and Retail Sales of Liquor and Alcohol [9],
* Walmart Dataset [10],
* Walmart Commerce Data [11],
* Soft Drink Sales [12].

Finalmente, se consultaron proyectos prácticos disponibles en **GitHub**, como Walmart-LSTM-Sales-Forecasting [13] y Retail-Demand-Forecasting [14], que permitieron observar la implementación real de modelos LSTM y variantes híbridas.

### ****2.2 Instrumentos Utilizados, Dinámicas Aplicadas y Alcance****

El Mapeo Sistemático de la Literatura se estructuró a partir de una metodología basada en criterios de inclusión y exclusión, con el fin de identificar fuentes relevantes y evitar sesgos.

**Criterios de inclusión:**

* Artículos publicados entre **2018 y 2025**.
* Aplicación directa en el dominio retail o e-commerce.
* Uso de redes neuronales LSTM, RNN o híbridos CNN-LSTM.
* Descripción explícita de datasets, procesos y métricas de evaluación (MAE, RMSE, MAPE).
* Acceso completo al texto.

**Criterios de exclusión:**

* Estudios fuera del dominio retail (energía, transporte, finanzas).
* Publicaciones sin validación experimental o métricas cuantitativas.
* Fuentes no verificables (notas de prensa, marketing, blogs no científicos).
* Duplicaciones de estudios o versiones preliminares sin revisión por pares.

La estrategia de búsqueda implementada en las bibliotecas digitales siguió la cadena de búsqueda:

(“retail demand forecasting” OR “sales prediction” OR “product demand”) AND (“LSTM” OR “deep learning” OR “time series”) AND (“AI” OR “machine learning” OR “hybrid model”).

El alcance de la revisión se concentró en investigaciones y desarrollos que pudieran **aplicarse a PyMEs**, priorizando aquellas que demostraron reproducibilidad, código abierto o disponibilidad pública de datos.

### ****2.3 Presentación de Datos Recabados****

El análisis de las fuentes permitió organizar la información en tres categorías:

1. **Modelos de predicción** empleados en el dominio retail.
2. **Datasets** utilizados para entrenar y validar modelos.
3. **Métricas** aplicadas para la evaluación del rendimiento.

Los artículos [1], [2] y [3] demostraron que los modelos basados en LSTM superan en precisión a los métodos estadísticos tradicionales (ARIMA, SARIMA), mientras que los enfoques híbridos (CNN-LSTM, BiLSTM, GRU) incrementan la capacidad de generalización en series temporales complejas.

En cuanto a las fuentes de datos, los **datasets de Kaggle** resultaron los más empleados por su estructura limpia, disponibilidad y documentación. Estos contienen información detallada de precios, categorías, promociones y fechas de venta, aunque la mayoría carece de variables externas (clima, feriados, campañas), lo que limita la robustez de los pronósticos.

Los proyectos de **GitHub** complementaron la evidencia empírica, mostrando implementaciones reales de modelos LSTM y CNN-LSTM aplicados a escenarios retail, con resultados comparables a los obtenidos en los estudios académicos.

### ****2.4 Presentación de Gráficos y Variables de Análisis****

Los estudios analizados coincidieron en el uso de un conjunto de **variables centrales**:

* Nombre y categoría de producto.
* Precio y variaciones de precios.
* Promociones activas o descuentos.
* Fecha de venta y frecuencia temporal.

Estas variables se modelaron como **series temporales multivariadas**, permitiendo que los algoritmos de aprendizaje profundo aprendieran dependencias temporales y patrones de comportamiento de consumo.  
Las métricas más utilizadas para la evaluación de modelos fueron **MAE (Mean Absolute Error)** y **RMSE (Root Mean Square Error)**, por su capacidad para medir la magnitud y la gravedad de los errores de predicción respectivamente.

Gráficamente, los resultados reportados por los estudios muestran que:

* Los **modelos LSTM puros** reducen el error promedio entre un 15% y 25% frente a métodos clásicos.
* Los **modelos híbridos CNN-LSTM** alcanzan reducciones adicionales del 5% al 10% en RMSE, aunque con mayor costo computacional.
* Las **variantes BiLSTM y GRU** ofrecen mejoras en estabilidad y convergencia, siendo útiles en datasets extensos o con ruido.

### ****2.5 Análisis de la Información****

El análisis conjunto de las fuentes académicas y prácticas permitió establecer que los **modelos LSTM** constituyen la base más sólida para el desarrollo de sistemas predictivos en retail, mientras que las **arquitecturas híbridas** presentan un potencial superior en contextos de alta variabilidad de datos.  
Sin embargo, su implementación requiere infraestructura computacional robusta y experiencia técnica, lo cual dificulta su adopción por parte de las PyMEs.

De esta manera, se confirma la **brecha tecnológica** identificada: mientras las grandes corporaciones (Intel, Amazon, SAP, Oracle) implementan soluciones predictivas avanzadas, el segmento minorista carece de herramientas accesibles que combinen predicción de demanda y personalización de ofertas.

El proyecto aquí propuesto busca llenar ese vacío mediante una **plataforma modular, abierta y escalable**, sustentada en las evidencias recopiladas y adaptada al contexto operativo de los pequeños y medianos comercios.

### ****2.6 Conclusiones del Relevamiento****

El relevamiento e investigación de mercado permitieron concluir que:

* Los modelos **LSTM y CNN-LSTM** son las técnicas más efectivas y documentadas para predicción de demanda en retail.
* Los **datasets públicos de Kaggle** constituyen fuentes confiables para el entrenamiento y la validación inicial de modelos.
* Existe una **tendencia creciente hacia modelos híbridos**, que combinan componentes convolucionales y recurrentes para mejorar la precisión.
* La mayor barrera para la adopción de IA predictiva en PyMEs es la falta de herramientas accesibles y de bajo costo.
* El enfoque **MLOps + SDD** ofrece un marco metodológico ideal para asegurar la trazabilidad, automatización y escalabilidad de los modelos, adaptándose tanto a entornos locales como cloud.

En síntesis, el análisis del mercado y la literatura especializada validan la **viabilidad técnica y la oportunidad comercial** del Asistente Inteligente de Ventas y Gestión de Inventario, cuyo diseño se fundamenta directamente en los resultados del Mapeo Sistemático realizado.

## **Capítulo III: Entorno y Dominio del Sistema de Información**

### ****3.1 Entorno del Sistema de Información****

El sistema se desarrolla dentro del **entorno del sector retail y PyME**, caracterizado por la alta rotación de productos, la variabilidad en la demanda y la necesidad de responder rápidamente a cambios de mercado.  
En este contexto, la **digitalización de los procesos de venta y gestión de inventario** se convierte en un factor estratégico para la competitividad. Sin embargo, gran parte de los pequeños y medianos comercios carecen de sistemas avanzados de análisis predictivo y dependen de registros manuales o planillas dispersas.

El entorno tecnológico actual ofrece plataformas robustas como **SAP Integrated Business Planning (IBP)**, **Oracle Retail**, **Amazon Seller Central** y **Mercado Libre**, que integran inteligencia artificial en sus procesos de predicción y gestión logística.  
No obstante, estas soluciones se encuentran **fuera del alcance económico y operativo** de las PyMEs, ya que requieren licencias costosas, servidores dedicados y personal especializado.

El **Asistente Inteligente de Ventas y Gestión de Inventario** se inserta en este entorno como una **alternativa accesible y escalable**, basada en herramientas de código abierto y en arquitecturas flexibles que pueden implementarse tanto de forma local como en la nube.  
Su diseño modular permite adaptarse a diferentes niveles de madurez digital, facilitando la adopción gradual de la inteligencia artificial en negocios que aún no cuentan con infraestructura avanzada.

Además, el proyecto se enmarca dentro del ecosistema tecnológico moderno de **aprendizaje automático aplicado**, empleando librerías y servicios como **TensorFlow**, **MLflow**, **Docker** y **Google Cloud Run**, lo que asegura portabilidad, trazabilidad y replicabilidad de los experimentos desarrollados.

### ****3.2 Dominio del Sistema de Información****

El dominio del sistema se centra en las operaciones comerciales relacionadas con la **gestión de ventas, control de inventario y análisis del comportamiento de compra de los clientes**.  
El modelo de inteligencia artificial implementado tiene como objetivo aprender patrones históricos de demanda y comportamiento de los consumidores a partir de series temporales multivariadas.

Dentro de este dominio, se distinguen dos tipos de variables:

* **Variables endógenas:** aquellas que dependen directamente de la dinámica interna del negocio y que el sistema utiliza para inferir patrones de comportamiento. Incluyen:
  + Volumen de ventas por producto.
  + Precios de venta unitarios y su variación temporal.
  + Cantidad de transacciones por día o semana.
  + Categoría y tipo de producto.
  + Promociones y descuentos aplicados.
* **Variables exógenas:** factores externos al negocio que pueden influir en la demanda y que, si bien no se integran en la versión inicial del sistema, se consideran relevantes para etapas futuras. Entre ellas:
  + Estacionalidad y períodos festivos.
  + Condiciones climáticas.
  + Tendencias de consumo global y eventos sociales.
  + Cambios macroeconómicos o de precios mayoristas.

El sistema trabaja principalmente con variables **endógenas**, dado que son las que se encuentran disponibles de forma directa en los registros del comercio. Sin embargo, la arquitectura propuesta se diseñó para ser extensible, permitiendo incorporar variables exógenas en etapas futuras y mejorar la precisión del modelo predictivo.

Desde el punto de vista analítico, el **modelo LSTM (Long Short-Term Memory)** constituye el núcleo del dominio, encargado de procesar las secuencias temporales y capturar dependencias a largo plazo.  
Las métricas **MAE (Mean Absolute Error)** y **RMSE (Root Mean Square Error)** permiten evaluar su desempeño cuantitativo, mientras que los principios de **MLOps** aseguran la trazabilidad de cada experimento y los lineamientos de **Spec-Driven Development (SDD)** garantizan la coherencia entre los requisitos funcionales y los componentes desarrollados.

### ****3.3 Alcance del Sistema de Información****

#### ****Alcance del Sistema****

El sistema tiene como propósito **procesar datos históricos de ventas y generar predicciones de demanda y ofertas personalizadas**, brindando una herramienta integral de apoyo a la gestión comercial.  
Las principales funcionalidades incluidas en su alcance son:

* **Carga automática de datos** desde archivos CSV o bases de datos locales.
* **Entrenamiento de modelos predictivos** basados en LSTM univariado y multivariado.
* **Registro de experimentos y métricas** a través de MLflow.
* **Generación de alertas de stock** ante posibles quiebres o excesos de inventario.
* **Sugerencias de promociones personalizadas** según el perfil de cliente.
* **Visualización interactiva** de métricas y resultados mediante un dashboard intuitivo.

Estas funciones se integran en un flujo automatizado de procesamiento, análisis y visualización, donde el usuario final puede operar el sistema sin requerir conocimientos técnicos avanzados.  
El entorno fue diseñado para ser **replicable, extensible y portable**, facilitando futuras mejoras como la inclusión de modelos híbridos, integración de factores externos y optimización logística.

#### ****Alcance del Proyecto****

El alcance del proyecto, en términos de desarrollo, comprende la **implementación del prototipo funcional** del asistente inteligente, incluyendo:

* Diseño del pipeline de datos.
* Entrenamiento de modelos predictivos y validación de resultados.
* Integración de MLflow y despliegue en Google Cloud Run.
* Elaboración de la documentación técnica y académica del sistema.

No se incluye en esta etapa la comercialización del producto, la integración con sistemas ERP externos ni la automatización completa del flujo de datos en tiempo real.  
Estas líneas se proyectan como **trabajos futuros**, una vez validado el prototipo en entornos de prueba controlados.

### ****3.4 Límites del Sistema de Información****

El sistema propuesto se encuentra acotado por una serie de **límites técnicos, funcionales y operativos**, que determinan su alcance dentro del dominio y sus restricciones actuales:

1. **Límites Técnicos:**
   * El rendimiento del modelo depende del volumen y calidad de los datos disponibles.
   * El procesamiento en tiempo real no se encuentra implementado en esta etapa inicial.
   * El sistema requiere conexión a Internet para el despliegue y monitoreo mediante Google Cloud Run.
   * Se identificó que un **modelo híbrido CNN+LSTM** ofrece un mejor desempeño en la predicción de demanda al combinar la detección de patrones locales (CNN) con la memoria temporal (LSTM).  
     Sin embargo, **por cuestiones de tiempo, recursos computacionales y complejidad de implementación**, su desarrollo no se incluye en esta versión inicial del proyecto, quedando previsto para futuras ampliaciones del sistema.
2. **Límites Funcionales:**
   * El sistema se enfoca exclusivamente en predicción de demanda y personalización de ofertas.
   * No abarca la gestión administrativa completa del negocio (facturación, contabilidad, logística).
   * Las recomendaciones se basan en patrones históricos y no incluyen variables contextuales externas (clima, eventos, tendencias sociales).
3. **Límites Operativos:**
   * El uso óptimo requiere personal con nociones básicas de gestión comercial y manejo de datos.
   * La precisión de los resultados puede variar según la frecuencia de actualización del dataset.
   * Los experimentos iniciales se ejecutan en entornos controlados antes de su validación en casos reales.

Estos límites no representan restricciones permanentes, sino **condiciones de diseño inicial** orientadas a garantizar la estabilidad del prototipo.  
En fases futuras, se prevé ampliar el sistema hacia la integración con bases de datos en tiempo real, incorporación de variables exógenas y adopción de modelos híbridos (CNN-LSTM) que incrementen la precisión y robustez de las predicciones.

## **Capítulo IV: Modelo de Negocios del Proyecto**

### ****4.1 Definición de Negocios****

El proyecto **Asistente Inteligente de Ventas y Gestión de Inventario** se concibe como una **plataforma tecnológica basada en inteligencia artificial**, orientada a la **predicción de demanda y la personalización de estrategias de venta** en el sector minorista.  
Su actividad principal consiste en desarrollar y ofrecer un sistema accesible, escalable y adaptable que permita a **pequeñas y medianas empresas (PyMEs)** optimizar su inventario, anticipar tendencias de consumo y mejorar la fidelización de sus clientes mediante ofertas personalizadas.

**Actividades principales:**

* Diseño, entrenamiento y despliegue de modelos de IA para predicción de ventas.
* Procesamiento automatizado de datos de inventario y ventas históricas.
* Generación de alertas inteligentes y sugerencias de reposición de stock.
* Recomendación de promociones personalizadas por cliente o categoría de producto.
* Visualización de métricas y predicciones a través de un dashboard interactivo.

**Mercado general:**  
El proyecto se inserta en el **mercado de soluciones tecnológicas para la gestión comercial y analítica de datos en retail**, un sector en rápido crecimiento debido a la transformación digital de los procesos empresariales.  
El público objetivo está conformado por:

* **Comercios locales y PyMEs**, que buscan herramientas predictivas económicas.
* **Supermercados regionales y mayoristas**, interesados en optimizar su reposición.
* **Emprendedores digitales y tiendas online**, que requieren sistemas adaptables y de bajo costo.

**Tecnología de diferenciación:**  
El valor diferencial radica en la **integración de modelos de IA accesibles y reproducibles (LSTM)** con metodologías **MLOps** para la automatización del ciclo de vida del modelo y **Spec-Driven Development** para la validación de funcionalidades.  
Esto permite ofrecer una solución de predicción avanzada y personalización de ofertas sin depender de licencias corporativas ni infraestructuras cerradas, algo poco común en el ecosistema de herramientas para PyMEs.

### ****4.2 Definiciones Estratégicas: Visión y Misión****

**Visión:**  
Ser una plataforma de referencia en el uso de inteligencia artificial accesible para la gestión comercial, impulsando la transformación digital de las PyMEs y promoviendo una adopción sostenible y ética de las tecnologías predictivas.

**Misión:**  
Brindar a los pequeños y medianos comercios una herramienta inteligente que combine predicción de demanda, gestión de inventario y personalización de ofertas, utilizando modelos de aprendizaje automático reproducibles y metodologías abiertas que garanticen su escalabilidad y transparencia.

### ****4.3 Análisis de Rivalidad Amplificada****

El análisis del entorno competitivo se realizó utilizando un enfoque de **rivalidad ampliada**, basado en los **cinco factores de Porter**, adaptados al contexto tecnológico del retail inteligente.

#### ****Competencia actual del mercado****

* **SAP Integrated Business Planning (IBP)** y **Oracle Retail Cloud** son los líderes del mercado corporativo, con soluciones de planificación de la cadena de suministro y predicción de demanda altamente integradas.  
  Sus principales limitaciones para el segmento PyME son los **altos costos de licencia**, la necesidad de infraestructura propia y la complejidad de personalización.
* **Amazon Seller Central** y **Mercado Libre Seller Hub** ofrecen funciones de análisis y predicción, aunque **restringidas a sus propios ecosistemas**, lo que limita su independencia y capacidad de expansión.
* **Odoo ERP** representa una alternativa más accesible, con módulos de inventario y ventas, pero sin capacidades nativas de predicción de demanda o personalización inteligente de ofertas.

#### ****Competidores potenciales****

* Startups de software analítico y de IA aplicada a retail (como **DataRobot** o **H2O.ai**) que podrían desarrollar versiones simplificadas de sus plataformas para PyMEs.
* Soluciones emergentes basadas en **AutoML** (Automated Machine Learning) que simplifican el entrenamiento de modelos predictivos sin requerir programación avanzada.
* Empresas de consultoría tecnológica que podrían ofrecer módulos específicos de predicción de ventas como complemento a sus servicios ERP.

#### ****Competidores sustitutos****

* Herramientas tradicionales de **análisis estadístico o BI** (Power BI, Tableau, Excel avanzado) utilizadas para generar reportes descriptivos.
* Aplicaciones de gestión de inventario que no incorporan IA, pero que ofrecen automatización parcial mediante reglas fijas o plantillas configurables.  
  Estos sustitutos resuelven parcialmente el problema, pero no proporcionan predicción ni personalización basadas en aprendizaje profundo.

#### ****Proveedores****

Los principales proveedores tecnológicos del sistema son:

* **Google Cloud Platform (GCP):** infraestructura para despliegue (Cloud Run) y almacenamiento escalable.
* **TensorFlow y Keras:** frameworks para el entrenamiento de modelos LSTM.
* **MLflow:** gestión de experimentos y versionado de modelos.
* **Kaggle y fuentes abiertas:** repositorios de datasets para entrenamiento inicial.

#### ****Clientes****

Los clientes potenciales son:

* **Pequeños comercios minoristas**, como kioscos, almacenes o tiendas locales.
* **Supermercados regionales y cadenas de distribución.**
* **Empresas con tiendas online** que necesitan pronosticar demanda y diseñar promociones personalizadas.
* **Desarrolladores y consultores independientes**, interesados en adaptar la plataforma a distintos entornos de negocio.

## **Capítulo V: Planificación del Proyecto**

### ****5.1 Definición de Iteraciones, Sprints o Etapas del Proyecto****

El desarrollo del proyecto se estructura en **doce fases secuenciales**, que articulan el proceso de investigación, desarrollo e implementación del sistema de predicción de demanda y personalización de ofertas mediante técnicas de inteligencia artificial.  
Cada fase se vincula con hitos específicos y entregables concretos, permitiendo una trazabilidad clara del avance y una organización adaptable a metodologías ágiles y al enfoque híbrido **MLOps–Spec Driven Development (SDD)**.

#### ****Fase 1 – Definición del Proyecto (Semanas 1 y 2)****

En esta etapa se plantea la idea base del sistema, su misión, visión y alcance, estableciendo la justificación técnica y el problema a resolver.  
Se realiza un relevamiento inicial del entorno PyME, identificando necesidades insatisfechas en la predicción de ventas y la gestión de inventario.  
Desde la perspectiva de Spec-Driven Development, se definen los primeros requerimientos narrativos (specs) que guiarán el desarrollo:

1. El sistema debe permitir la carga de datos de ventas.
2. Debe generar predicciones automáticas de demanda.
3. Debe ofrecer alertas de quiebre de stock.
4. Debe sugerir promociones personalizadas para cada cliente.

#### ****Fase 2 – Relevamiento Teórico y Estado del Arte (Semanas 2 y 3)****

Se realiza una revisión bibliográfica de los principales modelos predictivos aplicados al sector retail, con foco en redes neuronales LSTM y arquitecturas híbridas CNN-LSTM, además de los fundamentos de aprendizaje profundo, MLOps y Spec-Driven Development.  
Esta fase consolida el marco teórico del proyecto y determina las herramientas y tecnologías a utilizar (Python, TensorFlow, MLflow, Google Cloud Run, Docker).  
El resultado es un informe técnico-académico que establece los antecedentes y las bases conceptuales del trabajo.

#### ****Fase 3 – Mapeo Sistémico y Dominio del Proyecto (Semanas 3 y 4)****

Se aplican principios de pensamiento sistémico para comprender el entorno del proyecto, identificando actores, flujos de información, relaciones de retroalimentación y límites del sistema.  
A su vez, se implementa un Mapeo Sistemático de la Literatura (MSL) en bases académicas como IEEE Xplore, ACM Digital Library y Google Scholar.  
Esta búsqueda permite identificar estudios recientes y datasets abiertos relevantes para el dominio del retail.  
El resultado es un mapa conceptual del dominio y un inventario de fuentes de datos válidas para el entrenamiento de modelos.

#### ****Fase 4 – Análisis de Datos y Selección de Fuentes (Semanas 4 y 5)****

Se selecciona el dataset Online Retail y otros conjuntos complementarios (Walmart, bebidas, comercio minorista).  
Se realiza una primera limpieza manual, revisión de la estructura y un análisis exploratorio de datos (EDA) para garantizar la coherencia de las variables.  
En esta fase se diseñan los flujos de datos siguiendo el principio **Data Pipeline** de MLOps, preparando los datos para la etapa de modelado.

#### ****Fase 5 – Investigación en MLOps y Spec-Driven Development (Semanas 5 y 6)****

El objetivo de esta fase es estudiar y adaptar herramientas orientadas a la automatización del ciclo de vida del modelo (MLflow, Docker, Google Cloud Run) y a la especificación de funcionalidades (OpenAPI, testing automatizado).  
Se crea el repositorio de desarrollo bajo estándares de MLOps, estructurado con carpetas /data, /src, /models, /results, y se configura MLflow para el registro y versionado de experimentos.  
Desde la óptica SDD, se definen formalmente las especificaciones funcionales (specs) que regirán el desarrollo iterativo y se documenta la arquitectura conceptual del sistema.

#### ****Fase 6 – Presentación Inicial (Semana 6)****

Se realiza una primera presentación del avance ante la cátedra, exponiendo los fundamentos teóricos, la metodología de trabajo híbrida MLOps–SDD, los avances en la preparación del entorno de desarrollo y el cronograma de ejecución técnica.  
El objetivo de esta instancia es validar el enfoque metodológico antes de la implementación del modelo.

#### ****Fases 7–9 – Desarrollo del Modelo 1: Predicción de Demanda por Producto (Semanas 7, 8 y 9)****

En estas tres semanas se inicia la fase de desarrollo técnico del modelo.  
Se comienza con el preprocesamiento manual y automatizado de los datos de ventas, garantizando su limpieza, coherencia temporal y normalización de variables.  
Posteriormente, se desarrolla el primer modelo predictivo basado en LSTM univariado, entrenado con las series temporales de ventas por producto.  
Esta fase representa las etapas **Build–Train–Track** del ciclo MLOps, donde cada experimento se registra en MLflow con sus parámetros, métricas y versiones de dataset.  
El modelo se evalúa utilizando **MAE (Mean Absolute Error)** y **RMSE (Root Mean Square Error)**, generando visualizaciones comparativas y reportes técnicos.  
El entregable de esta fase es un modelo funcional reproducible que establece la línea base de desempeño del sistema predictivo.

#### ****Fase 10 – Desarrollo del Modelo 2: Variación de Productos por Cliente (Semanas 10 y 11)****

Se entrena un modelo LSTM multivariado, capaz de analizar relaciones entre productos adquiridos por cliente, considerando precio, categoría y cantidad.  
El objetivo es identificar patrones de compra combinada y fortalecer el módulo de personalización de ofertas.  
Durante esta etapa se aplica la fase **Integrate–Test** del flujo MLOps y se validan los resultados frente al modelo anterior, documentando cada experimento como una nueva versión en MLflow.

#### ****Fase 11 – Integración de MLOps y Arquitectura Cloud (Semanas 11 y 12)****

Se consolida la integración técnica del sistema, incorporando MLflow remoto, contenedorización con Docker y despliegue en Google Cloud Run.  
Desde la perspectiva de Spec-Driven Development, se definen y validan los endpoints de comunicación (API REST) que permiten la interacción entre módulos del sistema.  
Esta fase marca la transición del entorno local a un entorno productivo escalable, aplicando principios de **CI/CD (Integración y Despliegue Continuos).**

#### ****Fase 12 – Documentación, Evaluación y Cierre del Proyecto (Semanas 13 y 14)****

En la fase final se integran los resultados técnicos, gráficos, métricas y documentación en los capítulos del informe y la presentación final del proyecto.  
Se realiza la evaluación integral del sistema, analizando su precisión, escalabilidad y facilidad de uso.  
Se identifican las líneas futuras de mejora, entre ellas:

* Automatización completa del pipeline de datos.
* Incorporación de módulos de explicabilidad (XAI).
* Optimización de la infraestructura y monitoreo en producción.

### ****5.2 Descripción de Entregables****

Cada fase del proyecto genera un entregable verificable que demuestra el avance técnico y metodológico:

| **Fase** | **Entregable Principal** |
| --- | --- |
| 1 | Documento de definición del proyecto y requerimientos narrativos. |
| 2 | Marco teórico y selección de tecnologías base. |
| 3 | Mapeo sistémico, dominios y fuentes de datos. |
| 4 | Dataset limpio y flujo de datos diseñado. |
| 5 | Repositorio MLOps y documentación de especificaciones (specs). |
| 6 | Presentación de avances y validación metodológica. |
| 7–9 | Modelo LSTM univariado entrenado y validado (baseline). |
| 10 | Modelo LSTM multivariado y módulo de personalización. |
| 11 | Sistema integrado en la nube (Docker + Cloud Run + MLflow). |
| 12 | Informe técnico final, métricas, documentación y presentación. |

### ****5.3 Organización del Equipo del Proyecto****

El proyecto se desarrolla bajo modalidad **individual**, asumiendo el autor los roles de:

* **Investigador:** búsqueda bibliográfica, mapeo sistemático, definición de objetivos.
* **Desarrollador:** implementación de los modelos LSTM y flujo MLOps.
* **Analista de datos:** preprocesamiento y validación de datasets.
* **Documentador técnico:** elaboración de informes y presentación final.

El trabajo cuenta con **supervisión académica** por parte de la cátedra, que realiza el seguimiento metodológico, las revisiones y la validación de entregables.  
En etapas futuras se prevé la incorporación de colaboradores técnicos para ampliar la infraestructura y realizar pruebas con usuarios finales.

### ****5.4 Cronograma del Proyecto****

| **Semana** | **Actividad Principal** | **Entregable** |
| --- | --- | --- |
| 1–2 | Definición del proyecto, objetivos, misión y visión. | Documento Cap. I. |
| 3–4 | Mapeo sistémico, análisis de mercado y datasets. | Informe Cap. II. Articulo Conaiisi |
| 5–6 | Configuración técnica: MLflow, Docker, Cloud Run. | Repositorio técnico y presentación inicial. |
| 7–9 | Desarrollo del modelo LSTM univariado. | Modelo baseline y métricas MAE/RMSE. |
| 10–11 | Modelo LSTM multivariado y personalización. | Modelo validado y documentación técnica. |
| 12 | Integración con MLOps, API y despliegue en nube. | Prototipo funcional (MVP). |
| 13–14 | Documentación final y presentación. | Informe completo y cierre de proyecto. |

### ****5.5 Descripción del Producto Mínimo Viable (MVP)****

El **MVP** consiste en un prototipo funcional que permite:

1. Cargar datos de ventas históricos desde archivos CSV o bases de datos locales.
2. Preprocesar la información (limpieza, normalización, codificación).
3. Entrenar un modelo LSTM para la predicción de demanda por producto.
4. Generar métricas de validación (MAE, RMSE) registradas en MLflow.
5. Mostrar predicciones y alertas en un **dashboard interactivo**.
6. Desplegar el sistema en la nube mediante **Docker + Google Cloud Run**.

## **Capítulo VI: Metodologías de Gestión**

### ****6.1 Técnicas o Prácticas de la Ingeniería de Software para el Proyecto****

El proyecto se desarrolló bajo un enfoque **ágil, iterativo e incremental**, combinando los principios de la **Ingeniería de Software** con las prácticas emergentes de la **Ingeniería de Machine Learning (MLOps)** y la **Especificación Basada en Contratos (Spec-Driven Development – SDD)**.

Este enfoque híbrido permitió garantizar la reproducibilidad, escalabilidad y trazabilidad del ciclo de vida del sistema, desde la concepción de la idea hasta su despliegue en la nube.  
Las principales técnicas aplicadas fueron:

* **MLOps (Machine Learning Operations):** metodología derivada de DevOps adaptada a la ciencia de datos, utilizada para automatizar el ciclo de vida del modelo de IA. Se aplicaron sus etapas: Build–Train–Track–Integrate–Deploy–Monitor.
* **Spec-Driven Development (SDD):** enfoque basado en especificaciones narrativas y técnicas que definen, antes del desarrollo, los comportamientos esperados del sistema. Los specs sirvieron como contrato funcional para guiar las iteraciones.
* **Control de versiones con Git y GitHub:** permitió el registro histórico de los avances y la colaboración controlada, aplicando flujos feature-branch y pull requests como mecanismos de integración.
* **Prototipado evolutivo:** cada modelo (LSTM univariado, LSTM multivariado) se consideró un incremento funcional. Las mejoras se implementaron de manera incremental, preservando la estabilidad de versiones previas.
* **Integración Continua (CI/CD):** aplicada a través de scripts automatizados y contenedores Docker, asegurando que los cambios realizados pudieran ser probados y desplegados sin interferir en el sistema productivo.
* **Testing automatizado y validación de métricas:** se emplearon pruebas unitarias para componentes del pipeline (funciones de carga, preprocesamiento y evaluación de modelos), además de pruebas funcionales para verificar la generación de predicciones y visualizaciones.

Estas técnicas garantizaron que el desarrollo mantuviera la coherencia entre los requisitos definidos y los resultados alcanzados, favoreciendo la iteración continua y la mejora incremental del sistema.

### ****6.2 Trazabilidad del Proyecto****

La trazabilidad del proyecto se implementó mediante un sistema de **registro integral de actividades, versiones y métricas** que permitió seguir el progreso del desarrollo y vincular los entregables con sus fases correspondientes.

Las estrategias de trazabilidad aplicadas fueron:

* **Registro de experimentos en MLflow:** cada modelo entrenado (con sus hiperparámetros, dataset y métricas) fue almacenado como una run, permitiendo comparar versiones, reproducir resultados y verificar mejoras entre iteraciones.
* **Gestión documental:** los capítulos del proyecto, presentaciones y reportes se vincularon a las fases definidas en el plan de trabajo (Fases 1–12), asegurando la correspondencia entre documentación y avances técnicos.
* **Control de versiones de código:** Git y GitHub se utilizaron para mantener un historial claro de cambios en los scripts de entrenamiento, configuración del entorno, notebooks y documentación.
* **Trazabilidad de requisitos y specs:** cada especificación narrativa (SDD) se tradujo en una funcionalidad concreta validada al cierre de cada sprint. Por ejemplo, el spec “El sistema debe generar alertas de quiebre de stock” se implementó y verificó durante la fase del MVP.
* **Bitácora de avances semanales:** se documentaron los hitos alcanzados y los desafíos técnicos, proporcionando transparencia al seguimiento del proyecto académico.

De esta manera, la trazabilidad permitió verificar la alineación entre los objetivos planteados, los desarrollos realizados y los resultados finales obtenidos.

### ****6.3 Gestión de la Configuración del Proyecto****

La **Gestión de la Configuración (Configuration Management)** fue un componente esencial del proyecto, orientado a mantener la integridad técnica y la coherencia del sistema a lo largo de las diferentes fases de desarrollo.

Los elementos gestionados incluyeron:

* **Repositorio estructurado:** el proyecto se organizó bajo una jerarquía estandarizada:
* /data/ # datasets y preprocesamiento
* /src/ # scripts de entrenamiento y funciones auxiliares
* /models/ # versiones entrenadas de los modelos LSTM
* /results/ # métricas, gráficos y reportes
* /notebooks/ # análisis exploratorios y pruebas intermedias
* /mlruns/ # experimentos registrados por MLflow
* **Versionado de modelos:** cada modelo entrenado fue etiquetado con un identificador único (ej. LSTM\_v0.2, LSTM\_v0.3), almacenando su configuración, dataset y métricas de desempeño.
* **Configuraciones reproducibles:** el archivo requirements.txt documenta las librerías necesarias, mientras que el Dockerfile define el entorno exacto de ejecución para garantizar la portabilidad del sistema.
* **Gestión de entornos:** se establecieron tres entornos principales:
  + Development (entorno local de pruebas y ajustes).
  + Staging (entorno intermedio para integración y validación).
  + Production (despliegue final en Google Cloud Run).
* **Control de versiones de datasets:** se mantuvieron copias controladas de los datasets utilizados, asegurando que cualquier modelo pueda reproducirse bajo las mismas condiciones de entrenamiento.

Esta estructura de configuración permitió minimizar errores, evitar pérdidas de información y asegurar la estabilidad del proyecto en cada despliegue.

### ****6.4 Testing del Proyecto****

El **Testing** fue concebido como una práctica transversal en todo el ciclo de desarrollo, tanto en los componentes de software como en la validación de los modelos predictivos.

Se aplicaron los siguientes niveles de prueba:

* **Testing unitario:** validación de funciones de carga, limpieza y normalización de datos para asegurar consistencia en las transformaciones del pipeline.
* **Testing funcional:** verificación del comportamiento de los módulos principales: predicción de demanda, generación de métricas y visualización de resultados.
* **Testing de integración:** comprobación de la interacción entre componentes (modelo, API, dashboard) mediante endpoints definidos en los specs SDD.
* **Testing de desempeño:** evaluación de tiempos de entrenamiento, uso de recursos y estabilidad del sistema durante la ejecución de modelos en la nube.
* **Testing de métricas predictivas:** validación estadística mediante las métricas **MAE** y **RMSE**, garantizando que los errores de predicción se mantuvieran dentro de márgenes aceptables.

En una futura fase, se prevé la incorporación de pruebas automáticas de regresión sobre modelos entrenados, que permitan monitorear el rendimiento a lo largo del tiempo y detectar drift o degradación de desempeño.

El conjunto de pruebas aplicadas permitió garantizar que el sistema cumpliera con los requerimientos funcionales definidos y mantuviera la estabilidad a lo largo de sus iteraciones.

## **Capítulo VII: Marketing del Proyecto**

### ****7.1 Descripción del Producto / Experiencia****

El **Asistente Inteligente de Ventas y Gestión de Inventario** es una plataforma digital orientada a pequeños y medianos comercios (PyMEs) que integra herramientas de inteligencia artificial para la **predicción de demanda, la optimización del inventario y la personalización de ofertas comerciales**.  
Su valor diferencial radica en combinar funciones de análisis predictivo con automatización de decisiones comerciales, dentro de una solución accesible, escalable y fácilmente integrable en entornos de venta físicos u online.

Desde el punto de vista funcional, el sistema ofrece:

* **Predicción automática de ventas:** a partir del historial de transacciones y series temporales, el modelo LSTM anticipa la demanda futura por producto.
* **Gestión de inventario optimizada:** genera alertas ante posibles quiebres o sobrestock, priorizando productos según su rotación y rentabilidad.
* **Personalización de ofertas:** recomienda promociones y combos personalizados para cada perfil de cliente, en función de sus hábitos de compra y frecuencia de consumo.
* **Visualización interactiva:** a través de un dashboard intuitivo, el usuario puede observar métricas, tendencias y sugerencias generadas por el sistema.
* **Escalabilidad:** la arquitectura basada en contenedores (Docker + Google Cloud Run) permite su uso tanto en un kiosco individual como en una cadena de supermercados.

La experiencia de uso se caracteriza por su simplicidad técnica y su aporte estratégico: el sistema no solo automatiza tareas operativas, sino que **convierte los datos históricos en decisiones inteligentes**, mejorando la eficiencia, la fidelización de clientes y la rentabilidad del negocio.

### ****7.2 Descripción de las dinámicas de intercambio por perfil de usuario/interesado****

El proyecto contempla distintos perfiles de usuario e interesado, cada uno con dinámicas de interacción específicas dentro del sistema:

| **Perfil** | **Rol e Intereses Principales** | **Interacción con el Sistema** |
| --- | --- | --- |
| **Administrador / Dueño del negocio** | Busca optimizar la rentabilidad y reducir pérdidas por sobrestock o faltantes. | Accede al panel de control completo: consulta reportes de demanda, recibe alertas automáticas y visualiza indicadores de desempeño (MAE, RMSE). |
| **Encargado de ventas / Gestor de inventario** | Necesita información precisa para planificar reposiciones y gestionar pedidos. | Utiliza el módulo de predicción diaria/semanal y genera reportes de stock y rotación. |
| **Cliente final (usuario indirecto)** | Recibe beneficios mediante promociones personalizadas o descuentos relevantes. | Interactúa a través de la tienda o aplicación del comercio, recibiendo sugerencias y ofertas basadas en su comportamiento previo. |
| **Desarrollador / Equipo técnico (futuro)** | Interesado en integrar el sistema con otros servicios o expandir funcionalidades. | Accede a la API documentada mediante Spec-Driven Development, con endpoints definidos y escalables. |
| **Inversores o stakeholders externos** | Evaluan la viabilidad del producto y su impacto comercial. | Reciben informes de métricas, validaciones de modelo y proyecciones de mercado. |

Estas dinámicas aseguran que el sistema brinde valor a diferentes niveles: **operativo, analítico y estratégico**, promoviendo la adopción transversal del producto en distintos contextos de negocio.

### ****7.3 Estrategias de Publicidad y Promoción del Proyecto – Iniciativas de Engagement****

El plan de marketing y engagement se orienta a generar visibilidad, confianza y adopción progresiva del sistema dentro del ecosistema PyME y tecnológico.  
Las estrategias se estructuran en tres niveles: **posicionamiento digital, colaboración institucional y difusión experiencial**.

#### ****a) Posicionamiento Digital****

* **Creación de landing page informativa:** descripción del producto, beneficios, capturas de pantalla del dashboard y formulario de contacto.
* **SEO y marketing de contenido:** publicación de artículos técnicos y casos de uso sobre IA aplicada al retail, optimizados para motores de búsqueda.
* **Presencia en plataformas profesionales:** difusión del proyecto en LinkedIn, GitHub y Kaggle para atraer tanto usuarios potenciales como colaboradores técnicos.
* **Campañas segmentadas:** promoción dirigida en redes sociales (Instagram y Facebook) enfocada en comercios locales, con mensajes simples: “Predice tus ventas. Evitá pérdidas. Automatizá tu negocio.”

#### ****b) Colaboración Institucional****

* **Alianzas con cámaras de comercio y asociaciones PyME:** presentación del proyecto como solución tecnológica adaptable a distintos rubros.
* **Vinculación académica:** participación en eventos de investigación (CONAIISI, congresos de IA aplicada) para fortalecer la credibilidad científica y técnica.
* **Open Source Engagement:** publicación de versiones parciales del código y datasets para generar comunidad y fomentar la colaboración abierta.

#### ****c) Difusión Experiencial****

* **Pruebas piloto con comercios locales:** instalación controlada del sistema en entornos reales para demostrar su valor operativo.
* **Talleres y demostraciones:** capacitaciones breves para mostrar cómo interpretar las predicciones y usar las recomendaciones del asistente.
* **Testimonios y casos de éxito:** recopilación de experiencias de usuarios iniciales para fortalecer la confianza y la adopción del sistema.

#### ****e) Estrategias de Comercialización y Escalamiento Tecnológico****

Desde una perspectiva de negocio, el proyecto presenta oportunidades para su **escalamiento comercial y tecnológico**, incluyendo:

* **Modelo SaaS (Software as a Service):** ofrecer la plataforma como servicio en la nube, con suscripción mensual y distintos niveles de funcionalidad.
* **Licenciamiento a terceros:** integración del motor predictivo en plataformas de gestión ya existentes (ERP, CRM, POS).
* **Expansión regional:** adaptar la solución a distintos mercados latinoamericanos mediante localización de variables económicas y culturales.
* **Publicación Open Source controlada:** liberar una versión básica del sistema para fomentar la adopción y recibir aportes de la comunidad técnica.

Estas estrategias permitirían transformar el proyecto académico en una **solución tecnológica escalable**, con potencial de impacto económico y social en el sector PyME.

Estas iniciativas buscan **construir una relación sostenible entre tecnología y usuario final**, fomentando la comprensión de los beneficios de la inteligencia artificial aplicada al retail.

## **Capítulo VIII: Propiedad Intelectual**

### ****8.1 Definición de Marca y Activos a Proteger****

El proyecto **“Asistente Inteligente de Ventas y Gestión de Inventario”** constituye un desarrollo original que integra técnicas de inteligencia artificial, arquitectura cloud y analítica de datos aplicadas al sector retail.  
Por su carácter innovador, se identifican distintos activos intelectuales susceptibles de protección legal:

| **Activo** | **Descripción** | **Tipo de Protección Sugerida** |
| --- | --- | --- |
| **Nombre comercial y logotipo** | “Asistente Inteligente de Ventas y Gestión de Inventario” o una denominación abreviada (“SmartRetail AI”, “Predicta”) destinada a su difusión comercial. | **Marca** (denominativa y/o figurativa). |
| **Código fuente del software** | Scripts, funciones y modelos desarrollados en Python, junto con el pipeline de MLOps y las interfaces API documentadas bajo Spec-Driven Development. | **Derecho de autor / Software registrado** (Ley 11.723). |
| **Diseño de interfaz y dashboard** | Estructura visual del panel de control, disposición de métricas y visualizaciones. | **Derecho de autor / Diseño industrial (según caso)**. |
| **Modelos de IA y arquitectura MLOps** | Configuración y entrenamiento de modelos LSTM para predicción de demanda y personalización de ofertas. | **Secreto industrial / Documentación técnica confidencial.** |
| **Documentación técnica y metodológica** | Manuales, especificaciones funcionales (specs), plan de trabajo y reportes de métricas. | **Derecho de autor / Obra literaria técnica.** |

La protección de estos activos asegura la preservación del esfuerzo creativo y técnico del autor, evitando su reproducción o uso no autorizado por terceros.

### ****8.2 Definición de Clases Internacionales a Seleccionar****

Según la **Clasificación Internacional de Niza** (aplicable al registro de marcas en Argentina mediante el INPI), se identifican las siguientes clases adecuadas para la protección de los activos del proyecto:

| **Clase** | **Denominación según Clasificación de Niza** | **Aplicación al Proyecto** |
| --- | --- | --- |
| **Clase 9** | Software, programas informáticos descargables, aplicaciones móviles, inteligencia artificial. | Protege el software principal y sus módulos predictivos. |
| **Clase 35** | Servicios de análisis de datos, gestión comercial, consultoría empresarial. | Cubre los servicios de asesoramiento y predicción de demanda ofrecidos a empresas. |
| **Clase 42** | Desarrollo y diseño de software, servicios de computación en la nube, mantenimiento de sistemas. | Protege la plataforma técnica, el modelo de IA y la infraestructura cloud. |

En caso de expandir el proyecto hacia licenciamiento o comercialización, se recomienda el registro combinado en las clases **9, 35 y 42**, asegurando una protección integral tanto del producto digital como de su uso comercial y técnico.

### ****8.3 Resultados de Búsquedas en la Plataforma INPI****

## **Capítulo IX: Diseño de Solución**

### ****9.1 Arquitectura del Sistema de Información****

El **Asistente Inteligente de Ventas y Gestión de Inventario** se diseñó bajo una arquitectura modular, escalable y reproducible, compuesta por tres capas principales:

1. **Capa de Datos (Data Layer):** gestiona la adquisición, limpieza y transformación de la información.
2. **Capa de Modelado (Model Layer):** procesa los datos mediante redes neuronales LSTM para generar predicciones de demanda y sugerencias personalizadas.
3. **Capa de Aplicación (Application Layer):** expone los resultados al usuario final a través de una interfaz visual y servicios API definidos bajo Spec-Driven Development.

El diseño sigue el paradigma **“Data to Decision”**, donde los datos brutos se transforman progresivamente en información procesable y luego en decisiones comerciales automatizadas.

#### ****Flujo general de la arquitectura:****

1. **Entrada de datos:** el sistema recibe datasets estructurados (CSV, Excel o conexión a BD) con variables de ventas, precios y fechas.
2. **Preprocesamiento:** limpieza, normalización, codificación y enriquecimiento de datos con variables endógenas (ventas, precios, categorías) y exógenas (temporada, promociones, días festivos).
3. **Entrenamiento y validación:** los modelos LSTM son entrenados y evaluados mediante métricas MAE y RMSE, con seguimiento automático en MLflow.
4. **Predicción y visualización:** los resultados son almacenados en la base de datos del sistema y visualizados en un dashboard web.
5. **Despliegue:** el modelo y el dashboard se ejecutan en entornos cloud mediante contenedores Docker gestionados por Google Cloud Run.

Este diseño permite una **integración continua (CI/CD)** y una fácil escalabilidad del sistema para distintos tamaños de negocio, manteniendo la consistencia de versiones y resultados.

### ****9.2 Diseño de Solución****

El sistema se diseñó bajo principios de **MLOps** y **modularidad funcional**, donde cada componente cumple una función específica dentro del pipeline:

| **Componente** | **Función principal** | **Tecnologías empleadas** |
| --- | --- | --- |
| **Ingesta de datos** | Lectura y validación de archivos de ventas, productos y clientes. | Pandas, NumPy. |
| **Preprocesamiento** | Limpieza, normalización y codificación de variables categóricas. | Scikit-learn, Pandas. |
| **Modelado predictivo** | Entrenamiento de redes LSTM para predicción de demanda y patrones de compra. | TensorFlow/Keras. |
| **Gestión de experimentos** | Registro, versionado y comparación de modelos. | MLflow. |
| **Interfaz de usuario (UI)** | Visualización de resultados, métricas y alertas de stock. | Streamlit / Dash. |
| **Despliegue cloud** | Ejecución en contenedores y publicación de servicios. | Docker, Google Cloud Run. |

El diseño se implementa de manera **escalable y reproducible**, siguiendo estándares de codificación y documentación (PEP8, docstrings, comentarios estructurados).

### ****9.3 Modelo de Datos****

El modelo de datos se estructura en torno a tres entidades principales: **Producto**, **Cliente** y **Venta**, con relaciones que permiten analizar tanto la demanda global como los patrones de consumo individuales.

| **Entidad** | **Atributos principales** | **Descripción** |
| --- | --- | --- |
| **Producto** | id\_producto, nombre, categoría, precio, stock\_actual | Contiene la información básica de los artículos disponibles. |
| **Cliente** | id\_cliente, nombre, frecuencia\_compra, ticket\_promedio | Almacena datos del cliente y su comportamiento de compra. |
| **Venta** | id\_venta, id\_cliente, id\_producto, cantidad, fecha, promoción\_aplicada | Registra cada transacción con variables clave para análisis temporal. |

A partir de estas entidades se derivan vistas analíticas y series temporales agregadas por producto y cliente.  
El modelo se implementará en una base de datos **PostgreSQL o Supabase**, permitiendo consultas dinámicas y conexión directa con el modelo LSTM.

### ****9.4 Infraestructura de Comunicaciones y Servidores****

La infraestructura se diseñó para garantizar **eficiencia, portabilidad y seguridad**.  
Se compone de los siguientes elementos:

* **Servidor de datos:** almacena los datasets estructurados y las predicciones resultantes; implementado en **Google Cloud SQL** o Supabase.
* **Servidor de modelos:** aloja los modelos LSTM entrenados y gestionados mediante MLflow, con almacenamiento local y remoto sincronizado.
* **Servidor de aplicación:** despliega la interfaz (dashboard) y la API REST que permite la comunicación entre módulos.
* **Infraestructura Cloud:** el sistema se ejecuta en contenedores **Docker**, gestionados y escalados automáticamente por **Google Cloud Run**.
* **Seguridad y autenticación:** uso de claves API y restricciones de acceso por rol; en etapas futuras se incorporará OAuth2 para gestión de usuarios.

Esta infraestructura permite realizar predicciones bajo demanda, registrar resultados y escalar el servicio según la cantidad de usuarios o el tamaño del dataset.

### ****9.5 Entornos de Trabajo****

El proyecto se estructuró en tres entornos de trabajo coordinados, siguiendo las buenas prácticas de Ingeniería de Software:

| **Entorno** | **Propósito** | **Características principales** |
| --- | --- | --- |
| **Desarrollo (Local)** | Construcción y prueba inicial de componentes. | Python + MLflow local + Docker Desktop; base de datos SQLite. |
| **Staging (Preproducción)** | Integración y validación de módulos. | MLflow remoto + Google Cloud SQL; pruebas con datasets completos. |
| **Producción (Cloud Run)** | Ejecución del sistema funcional. | Despliegue Dockerizado en Google Cloud Run, monitoreo de métricas. |

Cada entorno está sincronizado mediante GitHub y MLflow, garantizando la reproducibilidad del ciclo de vida del modelo (ML Lifecycle Management).

### ****9.6 Solución****

El **Asistente Inteligente de Ventas y Gestión de Inventario** integra todas las capas descritas en una solución de software coherente, reproducible y adaptable a distintos contextos comerciales.  
Su arquitectura modular permite incorporar nuevos modelos, conectar bases de datos externas o integrar herramientas de visualización sin alterar la estructura base.

Los componentes clave de la solución son:

1. **Pipeline de datos:** flujo automatizado desde la carga hasta la limpieza y transformación.
2. **Motor predictivo LSTM:** núcleo del sistema encargado de generar pronósticos de demanda y detectar patrones de compra.
3. **API REST (Spec-Driven):** interfaz definida mediante OpenAPI para interactuar con el modelo y los módulos de visualización.
4. **Dashboard inteligente:** panel interactivo que presenta métricas, gráficos de tendencia y alertas de stock.
5. **Gestor MLOps (MLflow):** registro completo del ciclo de entrenamiento, validación y despliegue.
6. **Infraestructura Cloud:** ejecución distribuida mediante Docker y Google Cloud Run, con escalabilidad automática.

La solución propuesta demuestra la **viabilidad técnica y funcional** de aplicar modelos de IA avanzados al contexto PyME, ofreciendo una herramienta accesible, personalizable y con potencial de expansión futura hacia modelos híbridos **CNN-LSTM** y módulos explicables (**XAI**).

s identificadas en el modelo LSTM es su **baja interpretabilidad** para usuarios no técnicos.  
La incorporación de módulos de **IA Explicable (Explainable Artificial Intelligence – XAI)** permitirá:

* Justificar las recomendaciones y predicciones generadas.
* Aumentar la transparencia y confianza de los usuarios PyME.
* Facilitar auditorías de decisiones automatizadas y mejorar la usabilidad del sistema.

Esto podría lograrse mediante el uso de librerías como **SHAP** o **LIME**, integradas al pipeline de MLOps para explicar la influencia de cada variable en las predicciones.

#### ****c) Expansión de Fuentes de Datos y Contexto Exógeno****

Otra oportunidad clave radica en la **incorporación de variables externas (exógenas)** al modelo predictivo, tales como condiciones climáticas, calendarios de festividades, campañas publicitarias o tendencias en redes sociales.  
Estas variables pueden aumentar significativamente la precisión de las predicciones al capturar fluctuaciones estacionales o eventos externos que afectan la demanda.

Asimismo, se propone ampliar el uso de **fuentes de datos en tiempo real** mediante integraciones API con sistemas de facturación, ERP o plataformas de e-commerce.

#### ****d) Implementación de un Módulo de Optimización Logística****

Una extensión natural del sistema consiste en desarrollar un módulo adicional que vincule los resultados de la predicción con la **optimización de la cadena de suministro**, permitiendo:

* Calcular automáticamente los niveles óptimos de stock por producto.
* Sugerir rutas y volúmenes de reposición.
* Minimizar costos de almacenamiento y transporte.

Este módulo integraría técnicas de **programación lineal y heurísticas de optimización** con los resultados del modelo LSTM/CNN-LSTM, dando lugar a un sistema integral de apoyo a la toma de decisiones.

### ****11.2 Conclusión del Capítulo****

El análisis de oportunidades confirma que el **Asistente Inteligente de Ventas y Gestión de Inventario** posee un amplio margen de crecimiento y especialización futura.  
La implementación progresiva de modelos híbridos **CNN-LSTM**, la integración de técnicas de **IA explicable**, y la expansión hacia contextos **multifactoriales y logísticos** representan las líneas de desarrollo más prometedoras.

Estas mejoras no solo reforzarán la **precisión y robustez del sistema**, sino que consolidarán su **valor competitivo y su potencial de transferencia tecnológica**, posicionando al proyecto como una base sólida para futuros emprendimientos en inteligencia artificial aplicada al comercio minorista.