

Proyecto “MagikStock”

Presentación Final Capstone

Integrantes del proyecto

01 Catary Rodríguez

02 Cristian Núñez

03 Nicolás Moreno

Descripción del proyecto

Problema o Dolor

Ineficiencia en el control de Stock

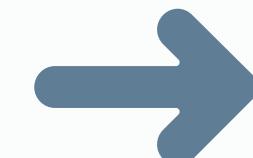
- Errores manuales causan desabastecimiento o exceso de productos.
- Dependencia de procesos manuales

Dificultad por crecimiento del negocio

- Debido a la expansión, la gestión del inventario se vuelve más compleja.

Monitoreo del Stock sin automatización

- Falta de actualización en tiempo real limita la respuesta a la demanda.



Propuesta de solución

Sistema automatizado

- Control de stock con monitoreo en tiempo real.

Notificaciones visuales

- Alertas claras con sistema de colores para niveles críticos de stock.

Unificación de sucursales

- Inventario centralizado con sincronización por ID para evitar errores.

Descripción del proyecto

Problema o Dolor

Ineficiencia en el control de Stock

- Errores manuales causan desabastecimiento o exceso de productos.
- Dependencia de procesos manuales

Dificultad por crecimiento del negocio

- Debido a la expansión, la gestión del inventario se vuelve más compleja.

Monitoreo del Stock sin automatización

- Falta de actualización en tiempo real limita la respuesta a la demanda.



Propuesta de solución

Sistema automatizado

- Control de stock con monitoreo en tiempo real.

Notificaciones visuales

- Alertas claras con sistema de colores para niveles críticos de stock.

Unificación de sucursales

- Inventario centralizado con sincronización por ID para evitar errores.

Dashboard predictivo

- *Análisis histórico y predicciones de tendencias para decisiones estratégicas, ofreciendo ventaja competitiva a Magikoffe.*

Objetivo General

Implementar un sistema automatizado de inventarios para reducir errores, mejorar precisión y apoyar decisiones estratégicas en Magikoffe.

Objetivos Específicos

- Sincronizar inventarios
- Monitoreo en tiempo real
- Notificaciones visuales dinámicas
- Dashboard predictivo
- **Seguridad basada roles**
- **Interfaz Intuitiva**

Alcances del proyecto

Entregables	Indicador de éxito
Sistema de Inventario en Tiempo Real	El numero de errores humanos en la gestión del inventario debe de reducir un 90% .
Sistema de alertas visuales	Las alertas deben ser visualizadas en menos de 30 segundos desde que el stock cae por debajo del umbral definido.
Dashboard Predictivo	La precisión de las predicciones de demanda debe ser superior al 75% .
Indicadores generales de éxito	<ul style="list-style-type: none">• Todos los entregables deben ser entregados a tiempo.• Cada entregable debe pasar al menos el 90% de las pruebas.

Premisas y restricciones del proyecto

Premisas

- El sistema actual de gestión de inventario se mantendrá operativo durante el desarrollo del nuevo sistema.
- Apoyo total de la gerencia para el desarrollo del nuevo sistema.
- **Escalabilidad del sistema:** nuevas sucursales.
- Adaptación al flujo de trabajo actual.
- Mantenimiento y soporte post-implementación.

Restricciones

- **Presupuesto limitado:** Presupuesto asignado por Magikoffee.
- **Plazo de implementación:** 16 semanas, ajustado a fases.
- **Requerimientos de seguridad:** confidencialidad de la información.
- **Interoperabilidad limitada:** Exclusivamente en el control de stock.

Metodología de trabajo para el desarrollo del proyecto

PMBOK 6^a edición



Inicio



Planificación



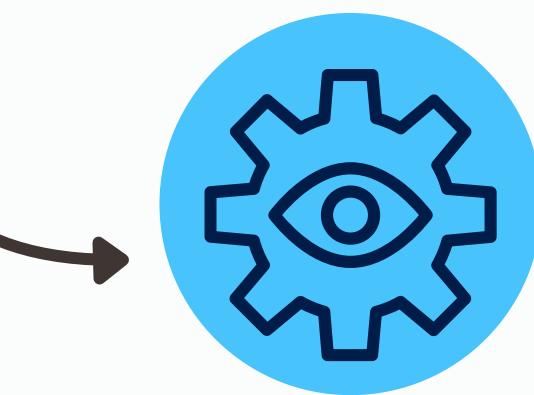
Ejecución



Monitoreo y control



Cierre



Ejecución y monitoreo

Cronograma para el desarrollo del proyecto

FASES	Inicio	Término	Mes n°1			
			Semana n°1	Semana n°2	Semana n°3	Semana n°4
Fase 1: Iniciación	12-agosto	18-agosto				
1. Desarrollar el Acta de Proyecto	12-agosto	18-agosto				
2. Identificar a los Stakeholders	12-agosto	18-agosto				
3. Definir los Objetivos y Alcance Iniciales	12-agosto	18-agosto				

Cronograma para el desarrollo del proyecto

FASES	Inicio	Término				Mes nº2	
				Semana nº3	Semana nº4	Semana nº1	
Fase 2: Planificación	19-agosto	8-sept					
4. Definir el Alcance	19-agosto	25-agosto					
5. Crear la EDT (Estructura Desglosada del Trabajo)	19-agosto	25-agosto					
6. Desarrollar el Cronograma Detallado	19-agosto	25-agosto					
7. Estimar Costos y Presupuesto	26-agosto	1-sept					
8. Planificar la Gestión de la Calidad	2-sept	8-sept					
9. Planificar la Gestión de Recursos	2-sept	8-sept					
10. Planificar la Gestión de las Comunicaciones	2-sept	8-sept					
11. Identificar y Analizar Riesgos	2-sept	8-sept					
12. Planificar la Gestión de Adquisiciones	2-sept	8-sept					

Cronograma para el desarrollo del proyecto

FASES	Inicio	Término	Mes nº1	Mes nº2			Mes nº3	
				Semana nº1	Semana nº2	Semana nº3	Semana nº4	Semana nº1
Fase 3: Ejecución	9-sept	20-oct						
13. Adquirir Recursos	9-sept	15-sept						
14. Asignar Roles y Responsabilidades al Equipo	9-sept	15-sept						
15. Diseñar la Arquitectura del Sistema	9-sept	15-sept						
16. Desarrollar Módulos y Funcionalidades	16-sept	7-oct						
17. Realizar Pruebas Unitarias y de Integración	30-sept	7-oct						
18. Corregir Errores Identificados	30-sept	7-oct						

Cronograma para el desarrollo del proyecto

FASES	Inicio	Término	Mes nº3		Mes nº4	
				Semana nº4	Semana nº1	
Fase 4: Ejecución y Monitoreo	21-oct	3-nov				
19. Implementar el Sistema (Despliegue en Ambientes de Prueba)	21-oct	27-oct				
20. Migración de Datos desde el Sistema Manual	21-oct	27-oct				
21. Monitorear Progreso y Seguimiento de KPIs	28-oct	3-nov				
22. Gestión de Cambios y Ajustes Necessarios	28-oct	3-nov				

Cronograma para el desarrollo del proyecto

FASES	Inicio	Término	Mes nº4			
			Semana nº1	Semana nº2	Semana nº3	Semana nº4
Fase 5: Monitoreo y Control	4-nov	17-nov				
23. Realizar Control de Calidad	4-nov	10-nov				
24. Revisar Cumplimiento de Requisitos	4-nov	10-nov				
25. Realizar Auditorías de Calidad	4-nov	10-nov				
26. Monitorear Riesgos Identificados	11-nov	17-nov				
27. Implementar Respuestas a Riesgos	11-nov	17-nov				
28. Generar Informes de Avance y Desempeño	11-nov	17-nov				

Cronograma para el desarrollo del proyecto

FASES	Inicio	Término	Mes nº1	Mes nº2	Mes nº3	Mes nº4		
								Semana nº4
Fase 6: Cierre	4-nov	17-nov						
29. Validar la Entrega Final con los Stakeholders	18-nov	24-nov						
30. Obtener Aprobación Formal del Sistema Implementado	18-nov	24-nov						
31. Compilar Manuales de Usuario y Documentación Técnica	18-nov	24-nov						
32. Realizar Reuniones de Retrospectiva	18-nov	24-nov						
33. Liberar Recursos y Cerrar Contratos	18-nov	24-nov						

Arquitectura de software

- La arquitectura del sistema de Magikoffee está diseñada siguiendo el enfoque del: **Modelo 4+1**
- Este modelo está basado en el uso de múltiples vistas que facilita la organización y gestión del sistema.
- Y en los escenarios, estos se componen de:

Vistas del Modelo

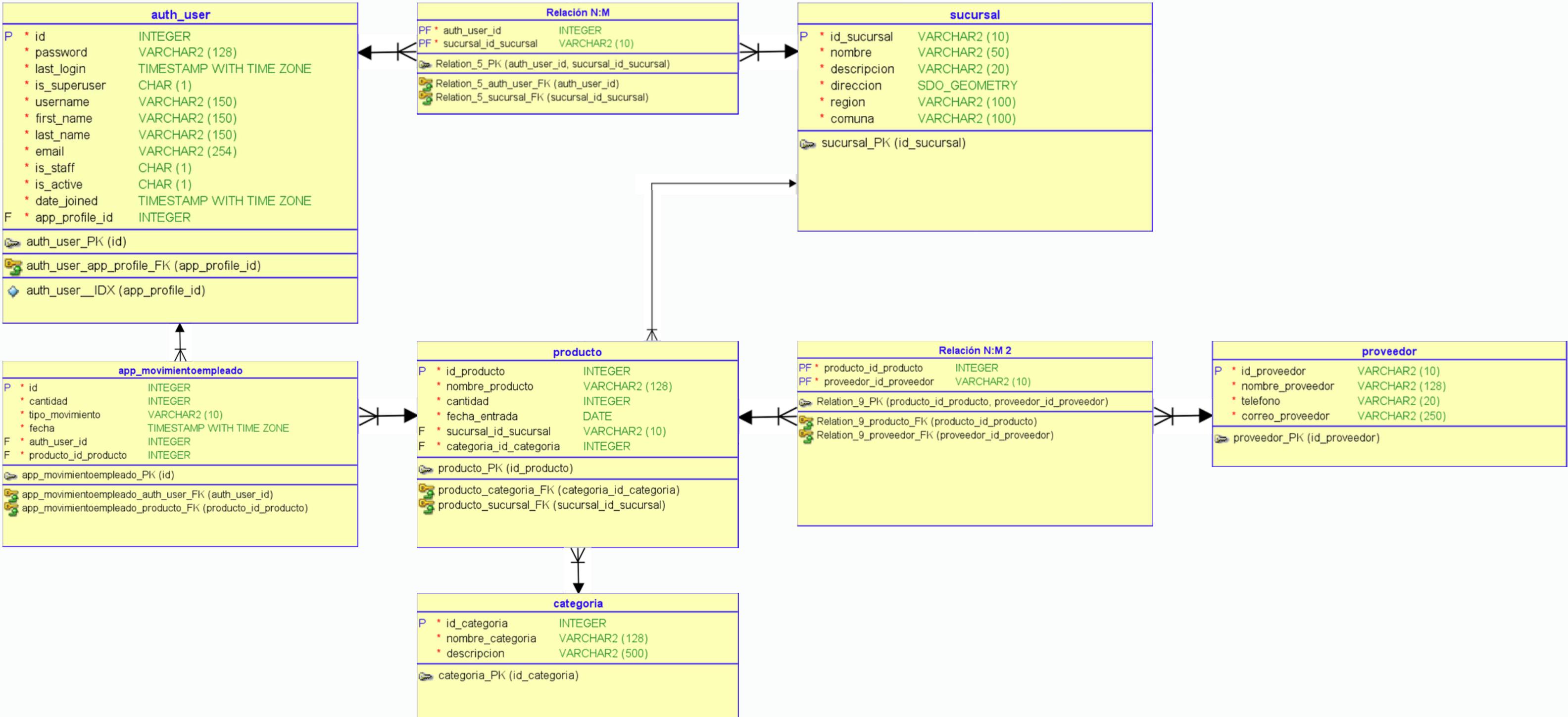
- **Vista lógica:** Estructura y funcionalidad del sistema.
- **Vista de desarrollo:** Perspectiva del programador.
- **Vista de proceso:** Explica los procesos y cómo se comunican.
- **Vista física:** Conexiones físicas entre estos componentes.

Casos de Uso y Escenarios de Calidad

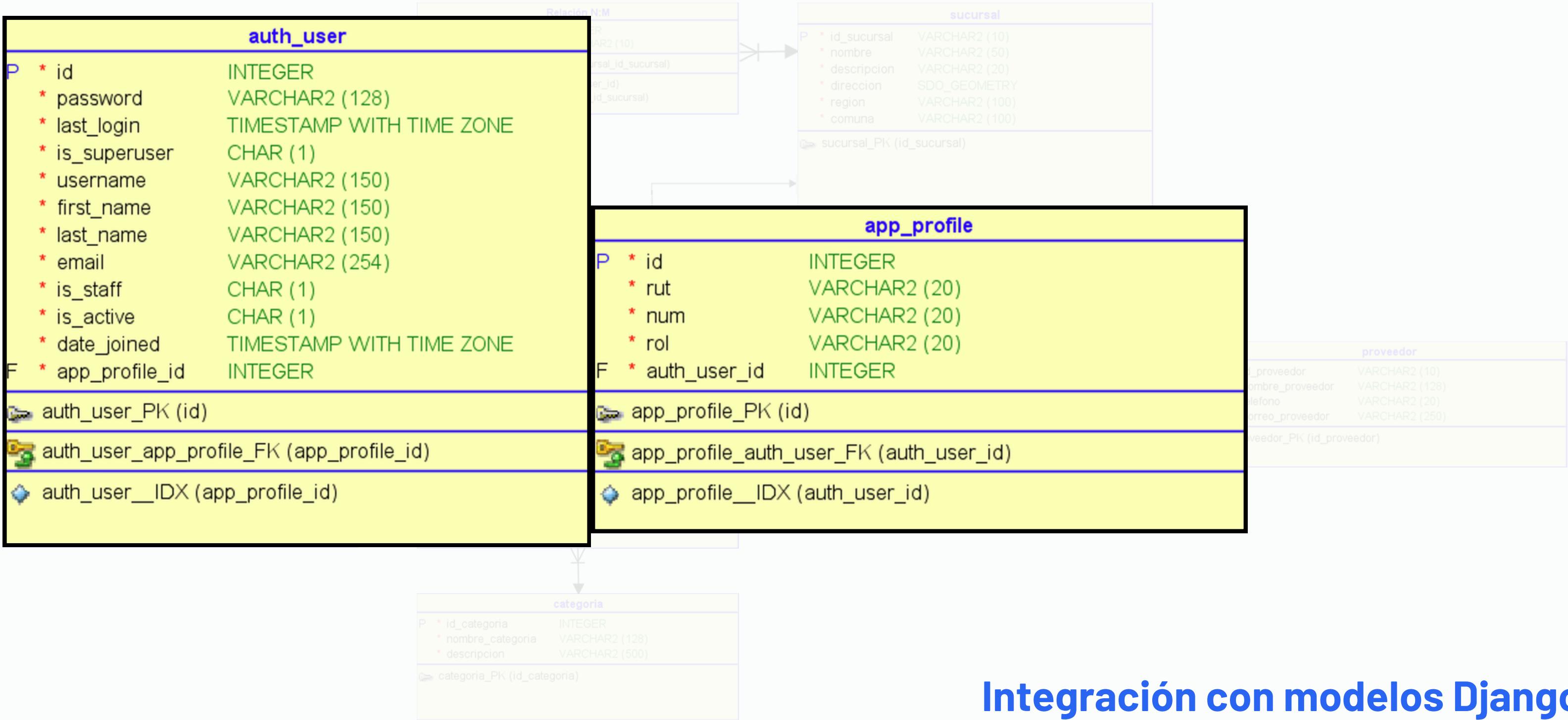
Los casos de uso describen funcionalidades específicas del sistema desde la perspectiva del usuario.

Y los escenarios de calidad describen atributos no funcionales del sistema tales como: rendimiento, seguridad, disponibilidad, etc.

Modelo de datos

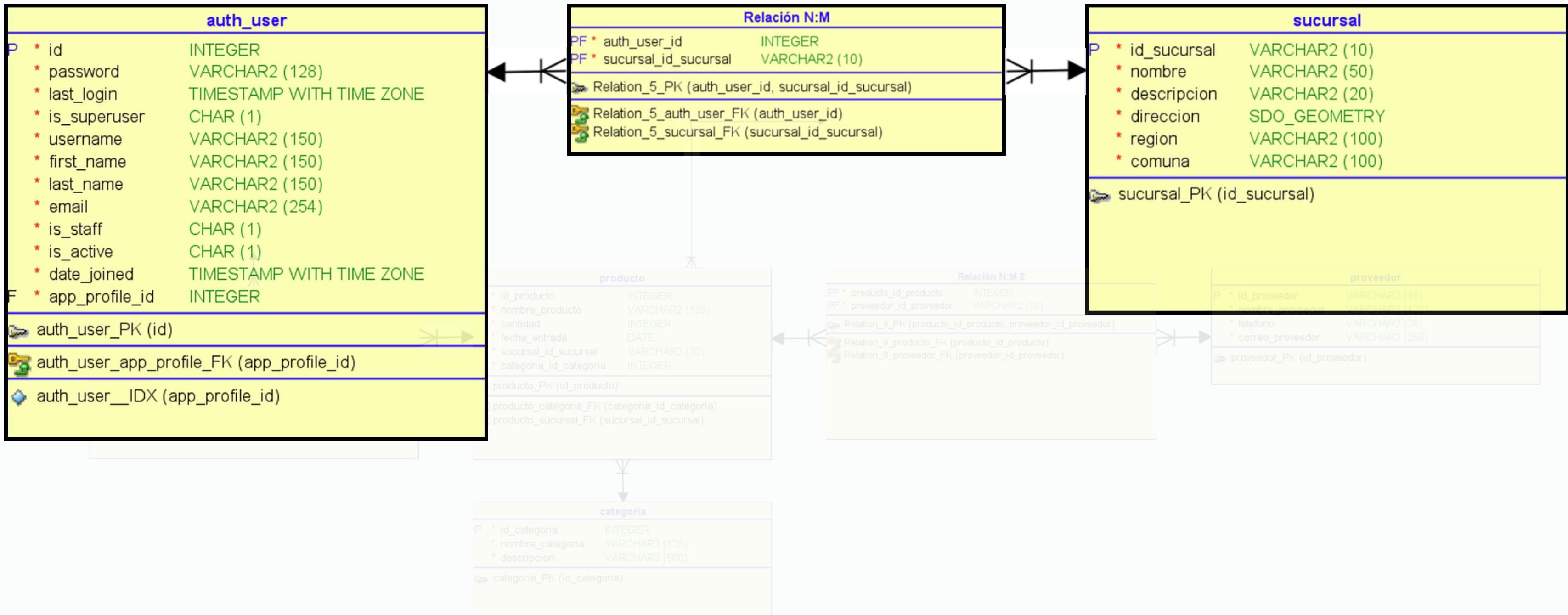


Modelo de datos



Integración con modelos Django

Modelo de datos



* username	VARCHAR2 (150)
* first_name	VARCHAR2 (150)
* last_name	VARCHAR2 (150)
* email	VARCHAR2 (254)
* is_staff	CHAR (1)
* is_active	CHAR (1)
* date_joined	TIMESTAMP WITH TIME ZONE
F * app_profile_id	INTEGER

auth_user_PK (id)

auth_user_app_profile_FK (app_profile_id)

auth_user_IDX (app_profile_id)

app_movimientoempleado

P * id	INTEGER
* cantidad	INTEGER
* tipo_movimiento	VARCHAR2 (10)
* fecha	TIMESTAMP WITH TIME ZONE
F * auth_user_id	INTEGER
F * producto_id_producto	INTEGER

app_movimientoempleado_PK (id)

app_movimientoempleado_auth_user_FK (auth_user_id)

app_movimientoempleado_producto_FK (producto_id_producto)



Modelo de datos

producto

P * id_producto	INTEGER
* nombre_producto	VARCHAR2 (128)
* cantidad	INTEGER
* fecha_entrada	DATE
F * sucursal_idSucursal	VARCHAR2 (10)
F * categoria_idCategoria	INTEGER

producto_PK (id_producto)

producto_categoria_FK (categoria_idCategoria)

producto_sucursal_FK (sucursal_idSucursal)

* region	VARCHAR2 (100)
* comuna	VARCHAR2 (100)

proveedor

P * id_proveedor	VARCHAR2 (10)
* nombre_proveedor	VARCHAR2 (128)
* telefono	VARCHAR2 (20)
* correo_proveedor	VARCHAR2 (250)

proveedor_PK (id_proveedor)

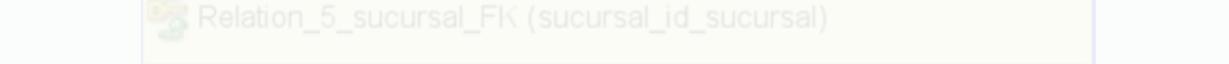


categoria

P * id_categoria	INTEGER
* nombre_categoria	VARCHAR2 (128)
* descripcion	VARCHAR2 (500)

categoria_PK (id_categoria)

* username	VARCHAR2 (150)
* first_name	VARCHAR2 (150)
* last_name	VARCHAR2 (150)
* email	VARCHAR2 (24)
* is_staff	CHAR (1)
* is_active	CHAR (1)
* date_joined	TIMESTAMP WITH TIME ZONE
F * app_profile_id	INTEGER
auth_user_PK (id)	
auth_user_app_profile_FK (app_profile_id)	



* region	VARCHAR2 (100)
* comuna	VARCHAR2 (100)
sucursal_PK (id_sucursal)	

Proyecto “MagikStock”

Modelo de datos

producto	
P * id_producto	INTEGER
* nombre_producto	VARCHAR2 (128)
* cantidad	INTEGER
* fecha_entrada	DATE
F * sucursal_id_sucursal	VARCHAR2 (10)
F * categoria_id_categoria	INTEGER
producto_PK (id_producto)	
producto_categoria_FK (categoria_id_categoria)	
producto_sucursal_FK (sucursal_id_sucursal)	

Relación N:M 2	
PF * producto_id_producto	INTEGER
PF * proveedor_id_proveedor	VARCHAR2 (10)
Relation_9_PK (producto_id_producto, proveedor_id_proveedor)	
Relation_9_producto_FK (producto_id_producto)	
Relation_9_proveedor_FK (proveedor_id_proveedor)	

proveedor	
P * id_proveedor	VARCHAR2 (10)
* nombre_proveedor	VARCHAR2 (128)
* telefono	VARCHAR2 (20)
* correo_proveedor	VARCHAR2 (250)
proveedor_PK (id_proveedor)	

categoria	
P * id_categoria	INTEGER
* nombre_categoria	VARCHAR2 (128)
* descripcion	VARCHAR2 (500)
categoria_PK (id_categoria)	



Tecnologías utilizadas

HTML



CSS



JS



Framework



Base de datos



Despliegue



Forecasting

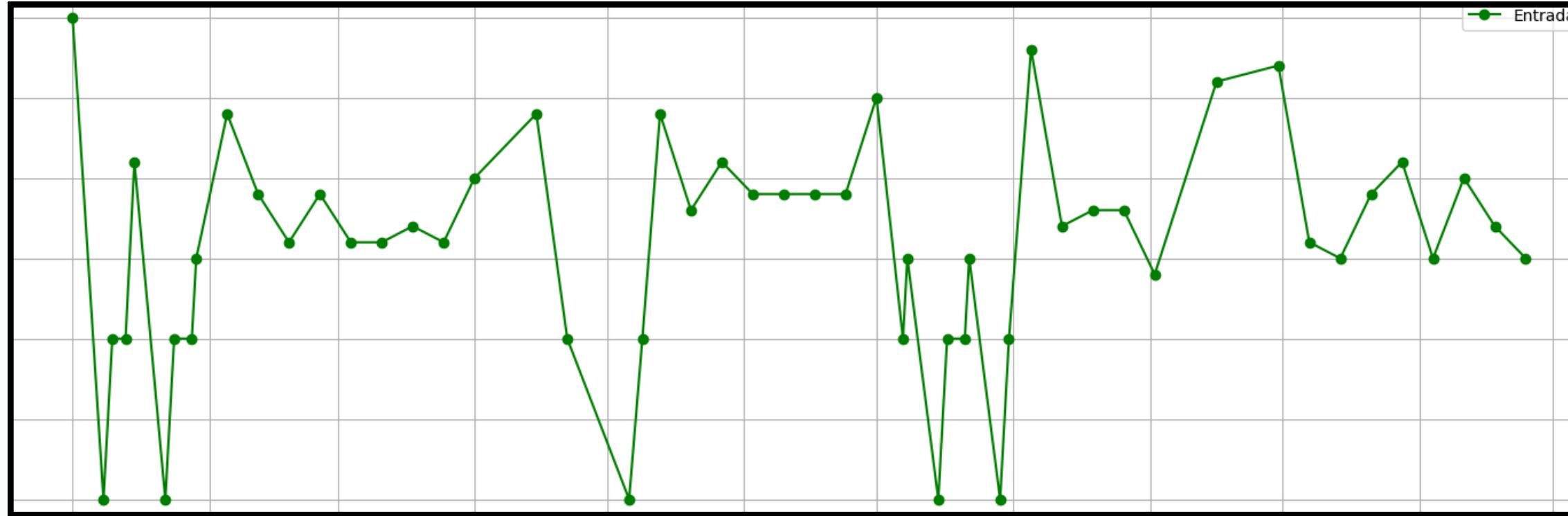


`!pip install pandas matplotlib prophet
from prophet import Prophet`

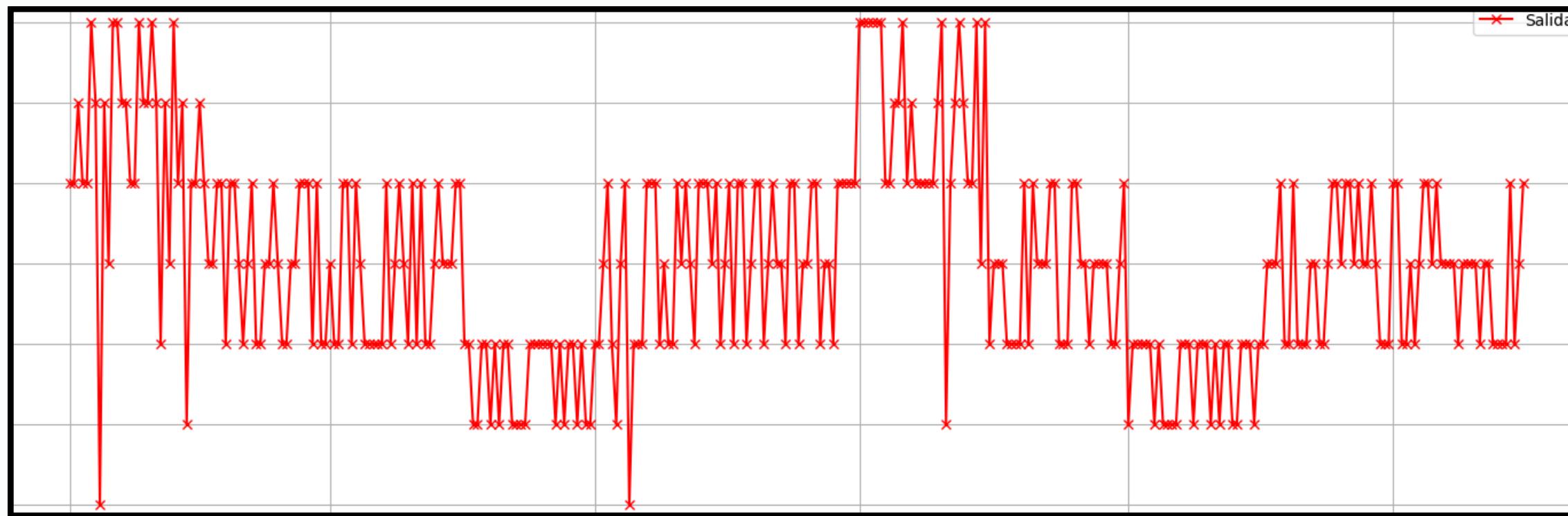
Dashboard Predictivo

Estimación del impacto futuro del stock

Movimiento de entradas

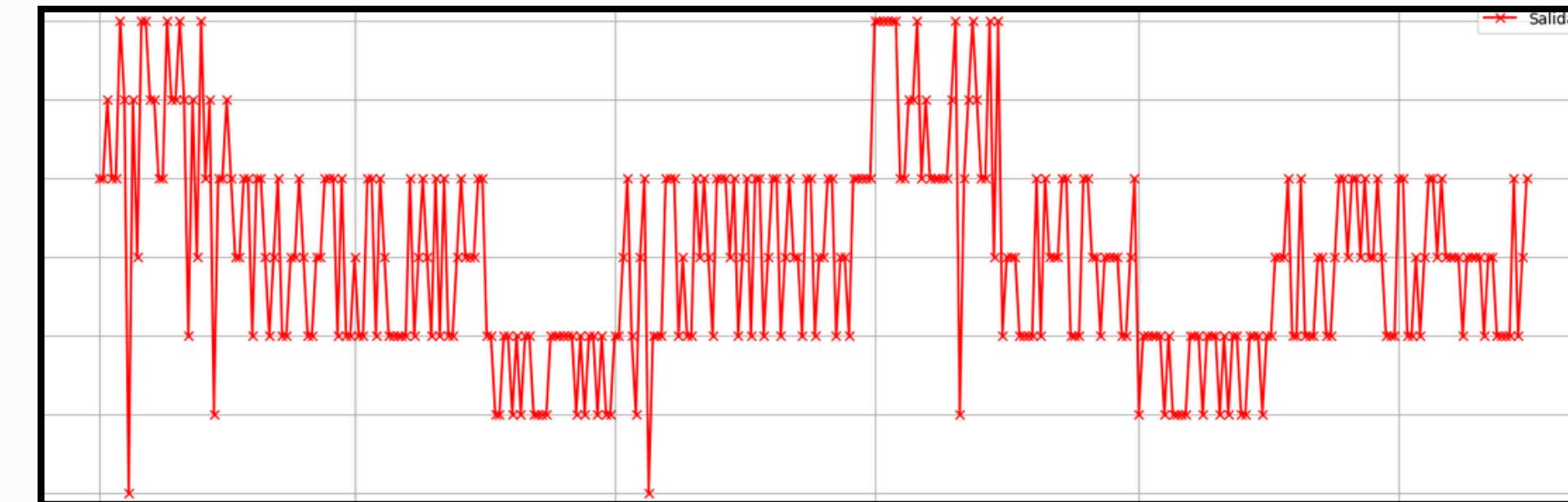
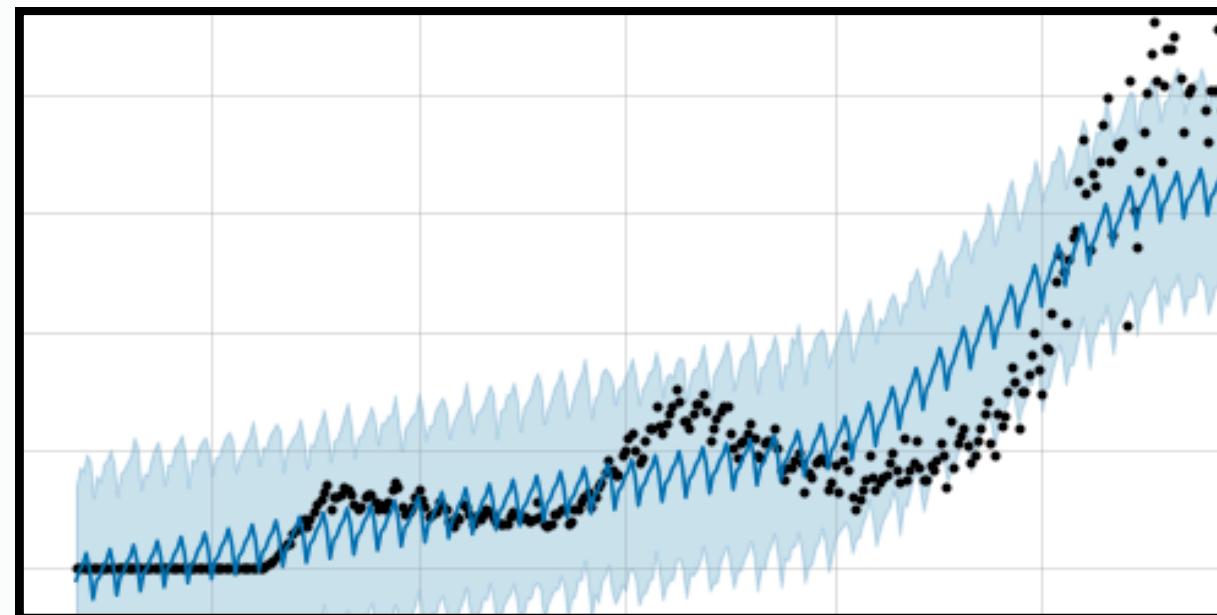


Movimiento de salidas



Forecasting mediante el Modelo Prophet.

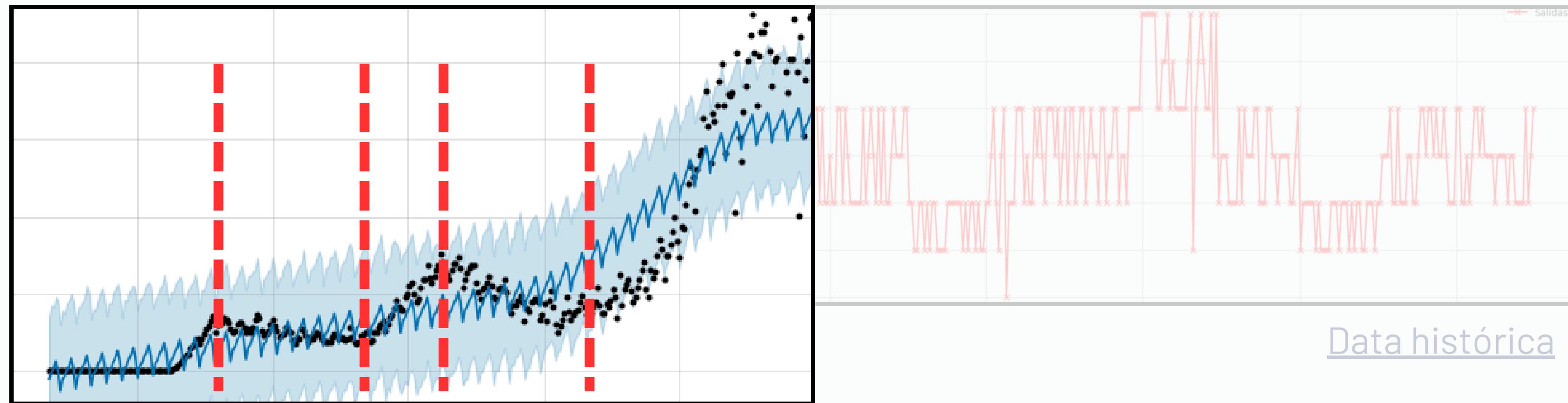
- Consiste en buscar la estimación y análisis de demanda **futura** mediante la data histórica.
- Prophet es un modelo desarrollado por **Meta**.
- Prophet busca **cambios de tendencias**.



Imágenes referenciales de: <https://towardsdatascience.com/>

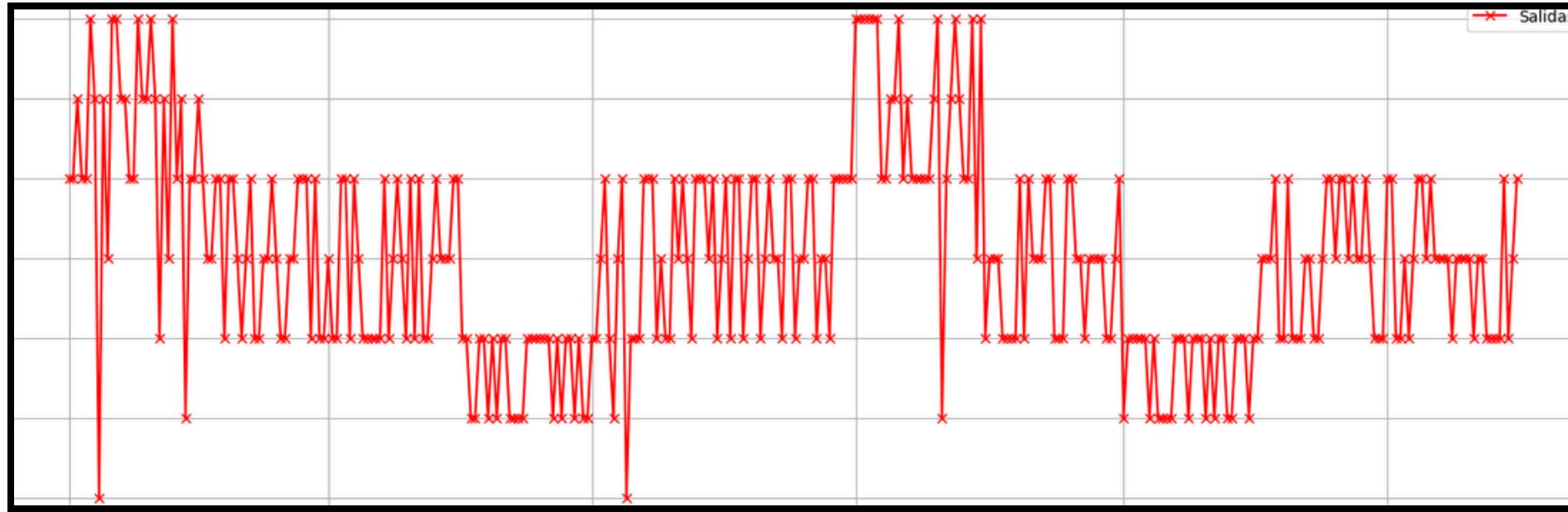
Forecasting mediante el Modelo Prophet.

- Consiste en buscar la estimación y análisis de demanda **futura** mediante la data histórica.
- Prophet es un modelo desarrollado por **Meta**.
- Prophet busca **cambios de tendencias**.



Imágenes referenciales de: <https://towardsdatascience.com/>

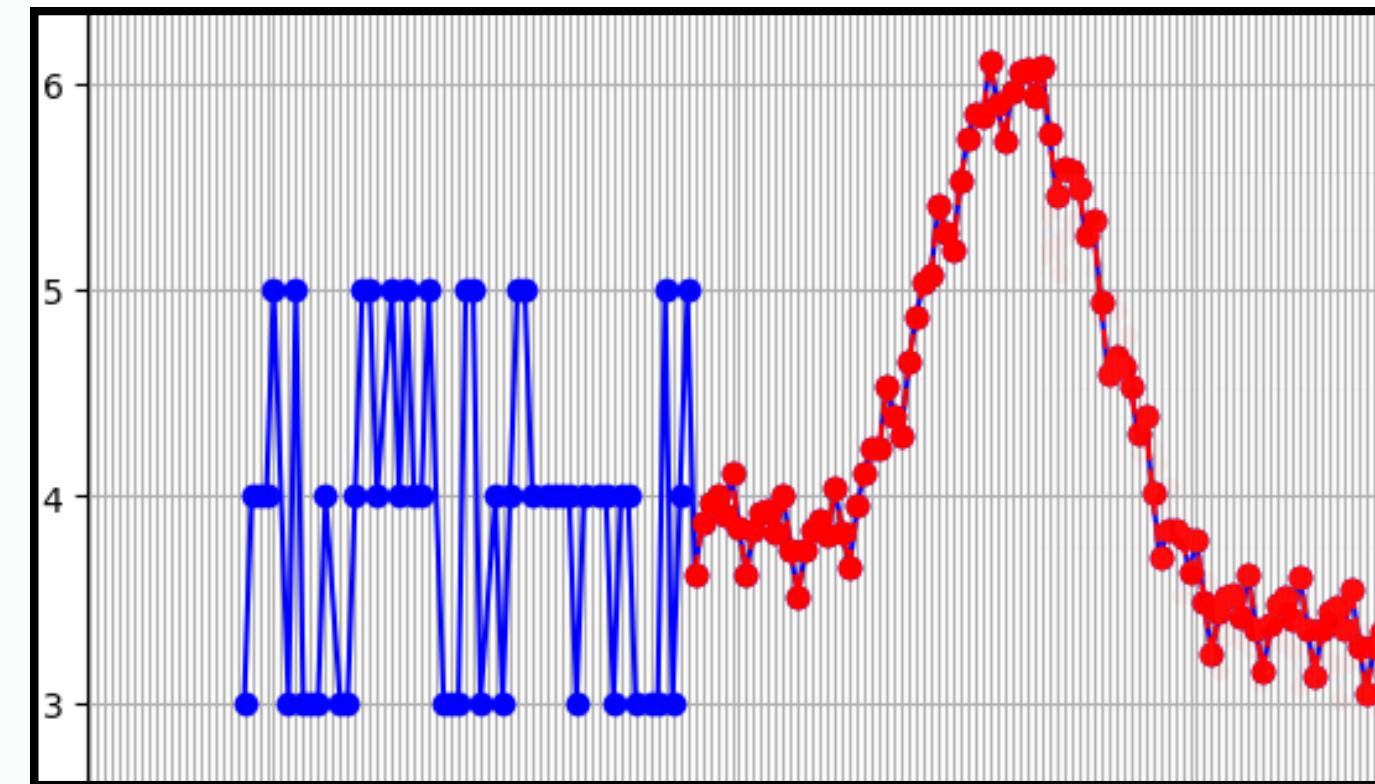
Data histórica



- Captura eventos regulares cada año, como picos estacionales: Invierno, verano.
- Identifica comportamientos basados en días de la semana.

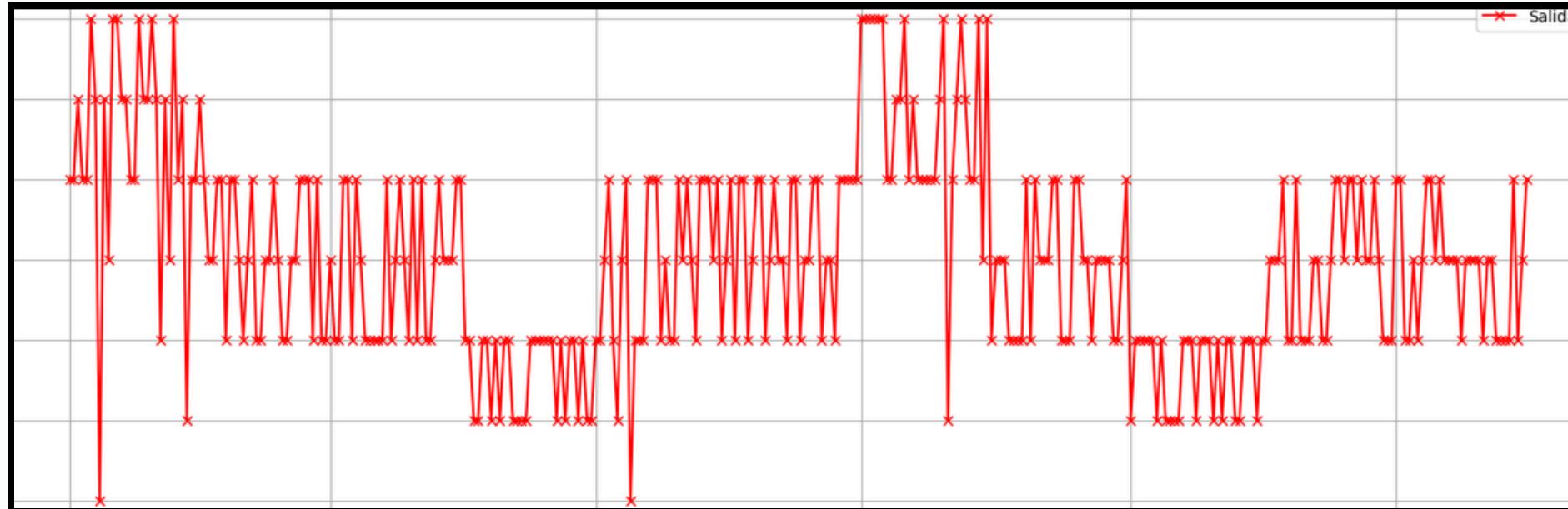
Parámetros principales:

1. **model** = Prophet(yearly_seasonality=True,
weekly_seasonality=True,
daily_seasonality=False)
2. **model.fit("data histórica")**
future = model.make_future_dataframe(
periods= "Días a predecir")
3. **forecast** = model.predict(future)

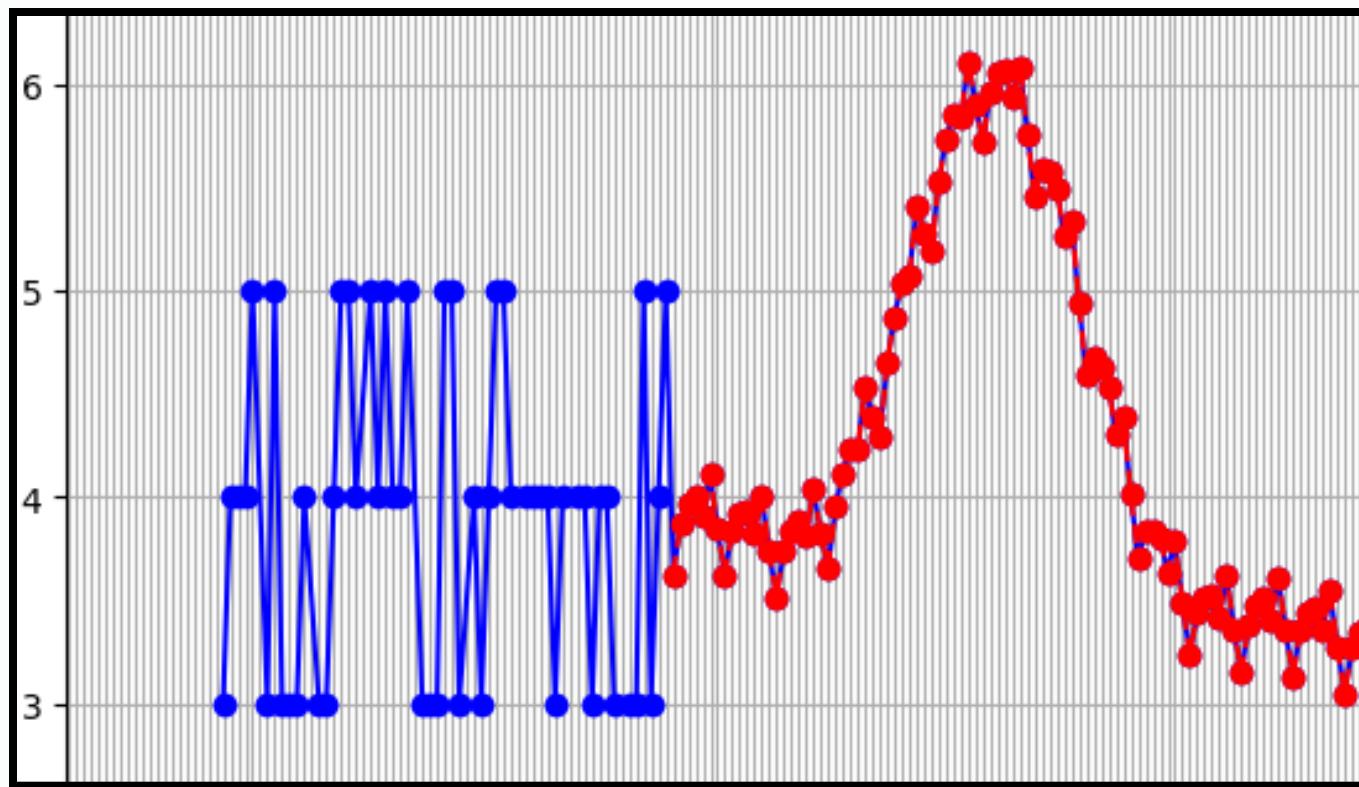


Predicción

Data histórica



- La data histórica mínima debe superar los 3 meses o los 90 registros.
- La predicción considerará un mes hacia el futuro.



Predicción

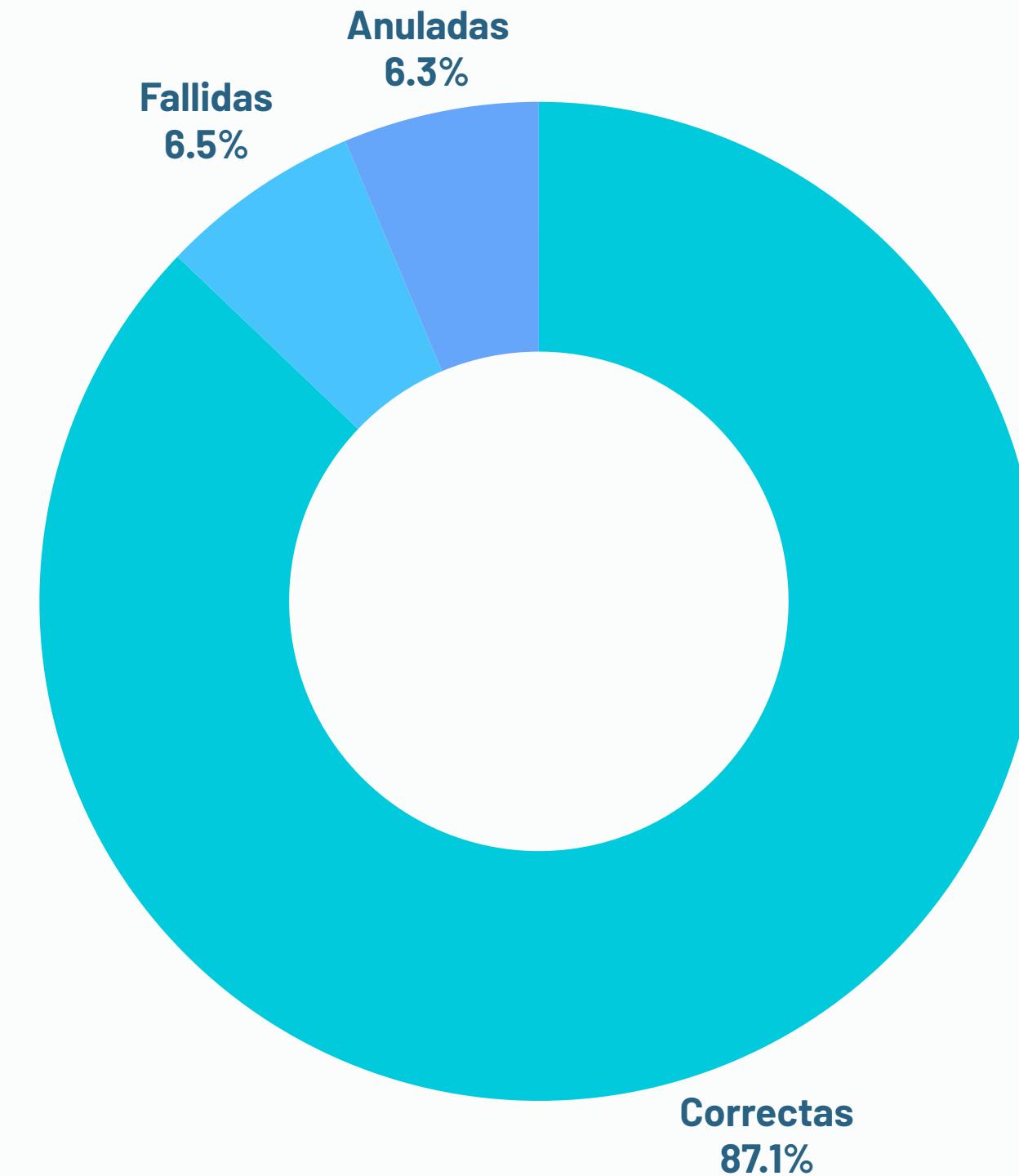
Resultados	
Menor a 90R	Promedio: 59.78%
90R - 365R	Promedio: 75.97%
Mayor a 365R	Promedio: 86,4% , ID119: 90.84%

Demostración del resultado del proyecto

<https://prototipov1-d23219a6f3e5.herokuapp.com/accounts/login/>

Resultados obtenidos

Casos de prueba		
Correcto	Fallido	Anulado
775	41	28
Casos de prueba estándar		
91	24	35
886	65	63
Total de pruebas: 994		



Resultados obtenidos

Alcances: Entregables del proyecto	Indicador de éxito	Resultado final
Sistema de inventario en Tiempo Real	Reducir un 90% los errores humanos.	<ul style="list-style-type: none">• 100% de efectividad en Casos de prueba.• Se disminuye de 5 errores promedio diarios (35) a 2 errores promedio semanales. Mejora del 94,2%.
Sistema de alertas visuales	Alertas visualizadas en menos de 30 segundos.	<ul style="list-style-type: none">• 100% de efectividad en Casos de prueba.• Las alertas son visualizadas en menos de 5 segundos.
Dashboard Predictivo	La precisión de las predicciones debe ser superior al 75%.	<ul style="list-style-type: none">• 94,6% de efectividad en Casos de prueba.• El promedio de predicción es del 86,4%. Con data insuficiente es en promedio superior al 75%.

Obstáculos presentados durante el desarrollo

- Pocas reuniones y demora en información relevante.
- Falta de tiempo para pruebas retrasó solución de fallos.
- Integración tomó más tiempo del esperado.
- Dificultades técnicas en la integración de componentes: Models.py - PostgreSQL
- Dificultades al conectar Django con PostgreSQL.
- Problemas con archivos estáticos afectaron el CSS en despliegue a Heroku.
- Digitalización de datos reveló errores, afectando el modelo predictivo

Lecciones aprendidas

¿Qué hicimos bien?

- Buena Comunicación.
- Planificación
- Mitigación de Riesgos
- Estimaciones de Recursos
- Retroalimentación del Usuario
- Diseño Escalable y Futuro

¿Qué podemos mejorar?

- Eficiencia Operativa del Equipo
- Pruebas Exhaustivas
- Claridad en los requisitos

Preguntas de la comisión

Muchas gracias