Autoevaluación Fase 1: StockFlow

Asignatura: Capstone - PTY4614(003V)

Autor: Matías Véliz, Alexander Seydewitz

2025

Contenido

[Resumen ejecutivo 2](#_Toc207732786)

[Descripción del proyecto 3](#_Toc207732787)

[Relación con las competencias del perfil de egreso 3](#_Toc207732788)

[Relación con intereses profesionales 3](#_Toc207732789)

[Factibilidad del proyecto 3](#_Toc207732790)

[Objetivos 4](#_Toc207732791)

[Objetivo general 4](#_Toc207732792)

[Objetivos específicos 4](#_Toc207732793)

[Metodología de trabajo 4](#_Toc207732794)

[Plan de trabajo 5](#_Toc207732795)

[Evidencias propuestas 5](#_Toc207732796)

[Conclusión 6](#_Toc207732797)

# Resumen ejecutivo

Se propone desarrollar una plataforma web de gestión de inventarios orientada a PYMES que permita registrar productos, actualizar existencias en tiempo real, gestionar precios y emitir alertas de stock bajo. Además, incluirá un módulo de análisis básico en Python que entregue recomendaciones sobre productos más vendidos y sugerencias para optimizar inventarios. La solución se implementará con Vite + Vue (frontend), Firebase (Firestore) como backend en tiempo real, y Python para análisis y generación de reportes. El proyecto integra competencias de desarrollo de software, modelado de datos y gestión de proyectos, y es factible dentro del marco de la asignatura por recursos, alcance y experiencia del equipo.

# Descripción del proyecto

Aplicación web que permite a administradores y encargados de inventario: registrar y editar productos, actualizar stock en tiempo real, configurar umbrales de alerta, visualizar reportes de ventas/stock y recibir recomendaciones generadas por un módulo analítico en Python (ventas por temporada, productos recomendados para reposición, sugerencias para optimización). La arquitectura contempla frontend en Vite + Vue, base de datos Firestore (Firebase) y scripts/servicios en Python para análisis y generación de reportes exportables.

## Relación con las competencias del perfil de egreso

El proyecto pone en práctica y evidencia las siguientes competencias del perfil de egreso definidas en la guía:

* Ofrecer propuestas de solución informática analizando procesos: Análisis de flujo de inventario y requerimientos funcionales.
* Desarrollar soluciones de software utilizando buenas prácticas de codificación: Uso de arquitectura modular, control de versiones, pruebas unitarias y E2E.
* Construir modelos de datos para soportar requerimientos: Diseño de esquema para Firestore.
* Programar consultas y rutinas para manipular información en bases de datos: Consultas para reportes, filtros y alertas en Firestore.

## Relación con intereses profesionales

Nuestros intereses se enfocan en desarrollo de software de calidad, diseño de arquitectura escalable y aplicación de análisis de datos para mejorar procesos empresariales. Este proyecto permite ejercitar: diseño arquitectónico frontend-backend, modelado de datos, integración con servicios en la nube (Firebase) y análisis en Python, consolidando competencias técnicas y experiencia práctica aplicable al mercado laboral.

# Factibilidad del proyecto

**Fortalezas que permiten factibilidad**:

* Conocimientos previos en Vue.js, Vite y fundamentos de análisis de datos.
* Documentación extensa y recursos disponibles de Firebase, Vite y librerías en Python.
* Equipo motivado y capacidad de trabajo colaborativo.

**Debilidades**:

* Experiencia limitada en modelos de recomendación → usar técnicas simples (frecuencias, análisis de series temporales básicas) y recursos online.
* Posibles complicaciones con sincronización en tiempo real → realizar pruebas incrementales, aprovechar la infraestructura de Firebase y documentar limitaciones.

Los recursos necesarios que tenemos disponibles es el entorno de desarrollo (VSCode), repositorio Git (GitHub), cuenta Firebase (Firestore), entorno Python (Jupyter/Colab), datasets simulados/CSV para pruebas. Estas condiciones hacen viable el desarrollo dentro del semestre/asignatura.

# Objetivos

## Objetivo general

Desarrollar una plataforma web funcional que permita la gestión en tiempo real de inventarios para PYMES y provea análisis y recomendaciones básicas para optimizar la reposición y ventas.

## Objetivos específicos

1. Diseñar e implementar la interfaz de usuario y las funcionalidades CRUD para productos y stock usando Vite + Vue.
2. Implementar la capa de persistencia en Firebase (Firestore) con manejo en tiempo real y reglas de seguridad básicas.
3. Desarrollar un módulo en Python que procese ventas/stock y genere reportes y recomendaciones simples (productos más vendidos por periodo, alertas de reposición).
4. Ejecutar pruebas unitarias e integración, y documentar evidencias (repo, demo, informe).
5. Desplegar una demo funcional (hosting) y preparar presentación final para evaluación.

# Metodología de trabajo

* Enfoque de desarrollo: Metodología ágil ligera (sprints de 3 semanas). Tareas en tablero (issues), entregas incrementales y revisión continua.
* Ciclo técnico: Planificación → diseño → desarrollo → pruebas → integración → despliegue.
* Pruebas y calidad: pruebas unitarias en frontend (componentes), pruebas de integración con Firestore, validaciones de entrada, revisión de código y uso de linters.
* Control de versiones y colaboración: GitHub con ramas por ambiente (desarrollo/certificación/producción)
* Análisis de datos: proceso inspirado en CRISP-DM para el módulo analítico (comprensión datos → preparación → modelado simple → evaluación → reporte). (Se aplicarán técnicas sencillas y explicables, dado el alcance).
* Documentación: README con instrucciones de instalación, manual de usuario, documentación técnica y registro de pruebas.

# Plan de trabajo

|  |  |
| --- | --- |
| Semana | Actividad |
| 1 | Definición de alcance, requisitos funcionales y no funcionales; creación repo. |
| 2 | Diseño UI y modelo de datos (Firestore). |
| 3 | Configuración Firebase y prototipo inicial frontend (esqueleto). |
| 4 | Implementación CRUD productos y gestión stock (frontend + Firestore). |
| 5 | Alertas de stock y reglas en Firestore; pruebas básicas. |
| 6 | Desarrollo del módulo de ingestión de datos (Python) y dataset de prueba. |
| 7 | Implementación de análisis (productos más vendidos, reportes) y API de conexión. |
| 8 | Integración total frontend ↔ análisis; generación de reportes exportables. |
| 9 | Pruebas unitarias, integración y corrección de errores. |
| 10 | Preparación de documentación, manual de usuario y evidencia. |
| 11 | Despliegue de demo (hosting) y grabación demo/video. |
| 12 | Presentación final, entrega de informe y ajustes según retroalimentación. |

*(Este proyecto cuenta con 4 principales Sprints de 3 semanas cada uno)*

# Evidencias propuestas

* Repositorio GitHub completo (código fuente, historial, ramas): Evidencia de desarrollo y control de versiones.
* Demo desplegada (URL o video de demo): Evidencia de funcionalidad y experiencia de usuario.
* Informe final: Descripción, metodología, resultados y lecciones aprendidas.
* Manual de usuario: Guía para operar la plataforma.
* Reporte de pruebas (unitarias e integración) y resultados de validación.
* Dataset y scripts de análisis en Python (Jupyter notebooks o scripts): Evidencia del componente analítico.
* Presentación demostrativa: Evidencia para evaluación oral y visual.

# Conclusión

El proyecto propuesto es pertinente, factible y alineado con las competencias del perfil de egreso. Permite fortalecer capacidades técnicas (desarrollo web, modelado de datos, análisis) y experienciales (gestión de proyecto, trabajo en equipo). Con una ejecución por fases, entregas incrementales y pruebas constantes, el alcance definido es realizable dentro del marco de la asignatura y permitirá generar evidencias claras para el portafolio de título.