

Universidad de las Fuerzas Armadas ESPE

Carrera: Ingeniería en Tecnologías de la Información

Tema: Control de Stock de Urban Style Shop

1. Información General

Asignatura: Metodología de Desarrollo de Software

Apellidos y nombres de los estudiantes: Kevin Cañola y Alexandro Molina

NRC: 20 967

Fecha de realización: 09/08/2025

Introducción

El presente informe detalla el desarrollo, seguimiento y cierre de tareas correspondientes al proyecto Urban Style Shop, específicamente en el módulo de Control de Inventario. Este proyecto se desarrolló bajo la metodología ágil Scrum, dividiéndose en sprints con objetivos claros, backlog de tareas priorizadas y asignaciones específicas para cada miembro del equipo.

Durante el periodo analizado, se llevaron a cabo actividades de planificación, ejecución, revisión y control de calidad, abarcando desde la categorización inicial de tareas hasta la entrega final de reportes de errores. La correcta priorización (importante, urgente, normal) permitió mantener un flujo de trabajo constante y cumplir con todos los plazos previstos, asegurando así la calidad y el avance del proyecto sin retrasos.

El registro analizado en este documento sirve como evidencia del compromiso del equipo, la distribución de responsabilidades y el grado de cumplimiento de los objetivos planteados para el módulo, garantizando que el producto entregado cumpla con los requisitos funcionales y técnicos establecidos.

Objetivo:

Objetivo general

Desarrollar un sistema de inventario en Java que permita a Urban Style Shop registrar, clasificar y controlar los productos disponibles, optimizando el manejo interno de stock y la consulta de historial de ventas.



Objetivos específicos

- Diseñar e implementar un sistema que permita a los administradores registrar y modificar productos de forma sencilla.
- Facilitar la clasificación de productos por categorías para una mejor organización visual y búsqueda eficiente.
- Generar reportes del inventario en formatos como PDF o Excel para llevar un control documental claro.
- Optimizar el control de stock mediante funciones que permitan aumentar o disminuir cantidades según necesidad.

Desarrollo

El conjunto de datos analizado corresponde a 17 tareas completadas dentro del marco del proyecto *Urban Style Shop* — módulo Control de Inventario. Todas las tareas registradas presentan estado COMPLETADO, lo que indica un cumplimiento total de los compromisos asumidos en el periodo reportado.

Clasificación por criticidad

Las tareas se organizaron por niveles de prioridad definidos desde la planificación del sprint:

- Importante (8 tareas): Actividades clave para el avance del proyecto, como la generación de backlogs con validaciones, revisión de análisis de requisitos y elaboración de reportes de errores.
- Urgente (3 tareas): Requerimientos que necesitaban atención inmediata, como categorización inicial de tareas, sincronización con calendarios y recordatorios de tareas incompletas.
- Normal (6 tareas): Actividades de soporte o continuidad, como generación de tareas, gestión de usuarios y revisiones técnicas.

La existencia de tareas urgentes, aunque en menor número, demuestra la capacidad del equipo para responder rápidamente a imprevistos sin comprometer los tiempos generales.

Cumplimiento de plazos

Uno de los hallazgos más relevantes es que ninguna tarea presentó retrasos. Esto se logró gracias a:

- 1. Planificación previa clara del backlog de cada sprint.
- 2. Asignación de responsabilidades precisas a los miembros del equipo.



3. Seguimiento constante mediante tableros Kanban, donde se actualizaba el estado de cada tarea.

En promedio, la duración de las tareas fue de 10 días, aunque este dato se ve influido por una tarea de gran envergadura: *Desarrollo del aplicativo Control de Inventario*, que tuvo una duración de 100 días. Excluyendo esta tarea, el promedio real de ejecución por tarea es de 1,5 días, lo que refleja una metodología ágil con entregas rápidas.

Distribución de responsabilidades

El reparto de tareas por integrante fue el siguiente:

- Kevin Cañola: 11 tareas (64,7% del total), asumiendo principalmente funciones técnicas como generación de backlogs, sincronización de calendarios y revisiones de requisitos.
- Alexandro Molina: 4 tareas (23,5%), concentradas en desarrollo del aplicativo, revisión de requisitos y gestión de usuarios.
- Jenny Ruiz: 3 tareas (17,6%), todas relacionadas con revisiones finales y reportes de errores.

Este patrón refleja que Kevin Cañola asumió la mayor carga operativa, mientras que Alexandro Molina aportó en el desarrollo de funciones clave y Jenny Ruiz en la verificación y control de calidad.

Flujo de trabajo

El avance de las tareas siguió una lógica de ciclo ágil:

- 1. Inicio → Categorización y generación inicial de tareas (SPRINT 1).
- 2. Planificación → Creación de backlogs y validaciones por sprint.
- 3. Ejecución → Programación, gestión y asignación de usuarios.
- 4. Revisión → Análisis de requisitos, sincronización y recordatorios de pendientes.
- 5. Cierre → Reportes de errores y verificación de requisitos completos.

Este orden evidencia un proceso iterativo que permitió entregar valor de forma continua al proyecto, garantizando que cada sprint cerrara con funcionalidades implementadas y revisadas.

Conclusiones

- Cumplimiento del 100% de los plazos indica una planificación y seguimiento efectivos
- La distribución de tareas muestra que Kevin Cañola ha asumido la mayor carga operativa, lo que podría requerir balancear responsabilidades para evitar sobrecarga.



- El orden lógico de ejecución de tareas (análisis, backlog, validaciones, revisiones y reportes) demuestra un flujo de trabajo alineado con buenas prácticas ágiles.
- La existencia de tareas urgentes sugiere capacidad de reacción rápida ante necesidades inmediatas.

Recomendaciones

- Equilibrar la carga de trabajo entre los miembros del equipo para evitar riesgos de dependencia de un solo integrante.
- Documentar lecciones aprendidas de cada sprint para mejorar la eficiencia y reducir la urgencia de tareas futuras.
- Implementar indicadores de desempeño (KPI) como *velocidad de sprint* y *tiempo de ciclo* para optimizar la planificación.
- Mantener la buena práctica de cero retrasos, reforzando la comunicación y seguimiento diario.

Enlaces

Git Hub

• https://github.com/AlexMolina21-21/MolinaAlexandro 20967 G5 METSW/tree/main/01 DOCUMENTACI%C 3%93N

Jira

• https://canola-met-desarrollo.atlassian.net/jira/software/projects/SCRUM/boards/1

Kanban

 https://github.com/AlexMolina21-21/MolinaAlexandro_20967_G5_METSW/blob/main/01_DOCUMENTACI%C3 %93N/PRESENTACION/Tablero%20Kanban%20-%20GRUPO%205_V3.xlsx

Cronograma

• https://github.com/AlexMolina21-21/MolinaAlexandro_20967_G5_METSW/blob/main/01_DOCUMENTACI%C3
%93N/PRESENTACION/Cronograma_Grupo_5_V7.xlsx

Enlaces videos

• https://github.com/AlexMolina21-21/MolinaAlexandro_20967_G5_METSW/blob/main/01_DOCUMENTACI%C3
%93N/PRESENTACION/ENLACES%20VIDEOS 4.pdf



Bibliografía

- Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. Scrum.org.
- Pressman, R. S. (2020). Ingeniería del Software: Un enfoque práctico (8^a ed.). McGraw-Hill.
- Sommerville, I. (2016). *Ingeniería de Software* (10^a ed.). Pearson Educación.
- PMBOK® Guide *Project Management Body of Knowledge* (7^a ed.). Project Management Institute.

Anexos

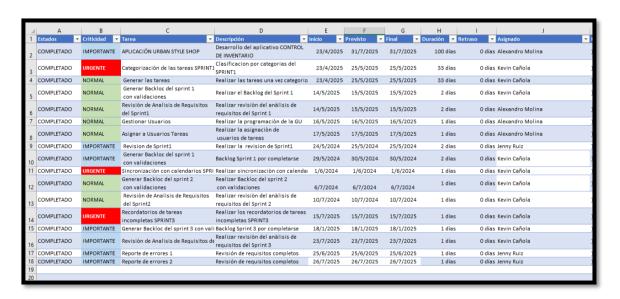
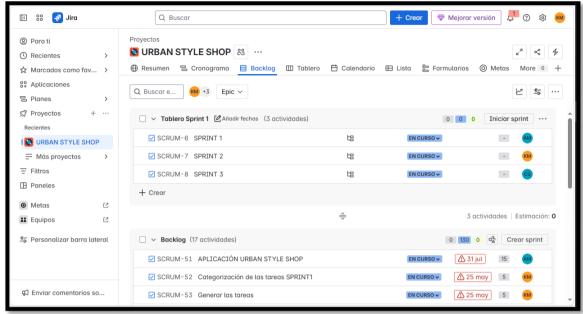
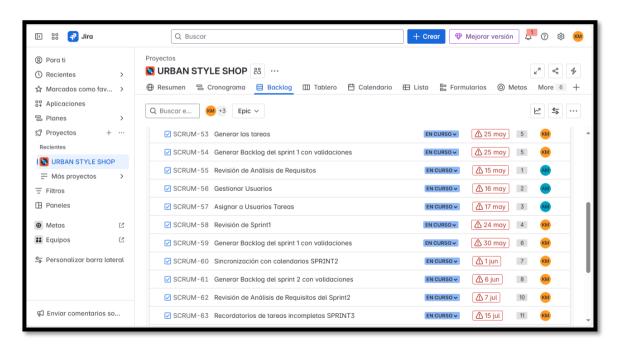


Tabla N° 1. − Kanban Excel



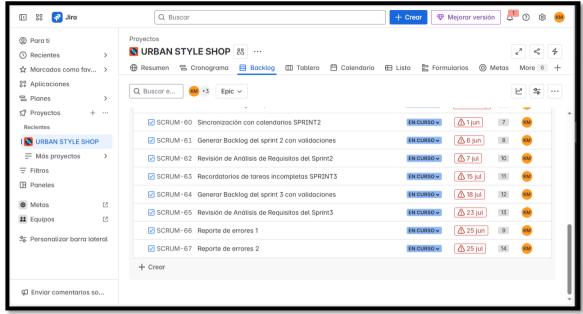


Registro Jira - SPRINT

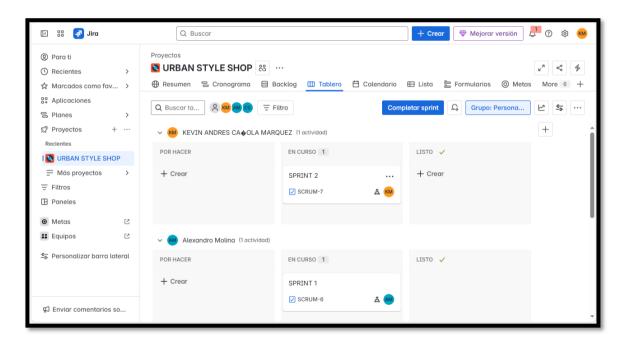


Registro Jira - Backlog



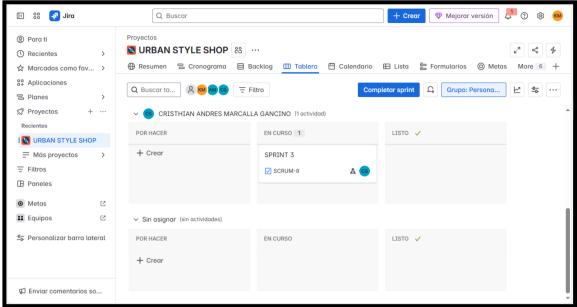


Registro Jira - Backlog



Registro Jira - Tablero





Registro Jira - Tablero