

# **SERVICIO NACIONAL DE APRENDIZAJE**

# **SENA**

**Actividad: GA1-220501093-AA1-EV04. Documento sobre la metodología para el proyecto de desarrollo de software**

**Programa:** Tecnología en Análisis y Desarrollo de Software

**Proyecto formativo:** Construcción de software integrador de tecnologías orientadas a servicios

**Fase:** Análisis

**Proyecto:** E-Commerce Sport+ S.A.S.

**Instructor:** Jhon Edison Nuñez Garzón

**Aprendiz:** Julián David Macías Garcés

**Fecha:** 14 de diciembre de 2025

# **Tabla de Contenido**

1. Resumen Ejecutivo
2. Introducción
3. Contextualización del Proyecto Sport+ S.A.S.
4. Marco Metodológico Seleccionado: Scrum
5. Justificación Detallada de la Metodología
  - 5.1. Contraste: Ágil (Scrum) vs. Tradicional (Cascada)
  - 5.2. Gestión de Riesgos y RNF Críticos
6. Estructura de Roles, Artefactos y Herramientas
  - 6.1. Roles Fundamentales
  - 6.2. Artefactos de Gestión
  - 6.3. Herramientas de Apoyo (Tooling)
7. Plan de Actividades Detallado bajo Scrum
8. Conclusión
9. Referencias Bibliográficas

# 1. Resumen Ejecutivo

Este documento formaliza la selección del marco de trabajo **Scrum** para la fase de diseño, construcción y pruebas del proyecto **E-Commerce Sport+ S.A.S.** La elección se fundamenta en la alta **volatilidad del mercado digital** y la necesidad imperativa de **mitigar riesgos** asociados a la usabilidad (RNF-01) y la seguridad (RNF-07) de la plataforma. A diferencia de los modelos predictivos (ej. Cascada), Scrum garantiza la **entrega de valor incremental** y la **adaptabilidad** a los requisitos cambiantes. La implementación se sustentará en roles definidos (Product Owner, Scrum Master), artefactos clave (Product Backlog) y herramientas de alto desempeño (Jira, Git, CI/CD) para asegurar un producto final alineado con la estrategia de negocio.

## 2. Introducción

La Fase de Análisis del proyecto Sport+ S.A.S. ha culminado con la definición de la **Especificación de Requisitos del Software (ERS)**. El siguiente paso crítico es la selección de un marco metodológico que permita transicionar de manera eficiente del análisis al diseño y la construcción. Una metodología robusta es el factor determinante para reducir la exposición a riesgos, asegurar la **trazabilidad** de los requisitos y garantizar la **calidad** del *software*. El presente informe justifica la aplicación de Scrum, detallando su estructura operativa y su plan de integración con el ciclo de vida del proyecto.

## 3. Contextualización del Proyecto Sport+ S.A.S.

El proyecto consiste en el desarrollo de una plataforma *E-Commerce* orientada a servicios para la venta de artículos deportivos.

- **Objetivo de Negocio:** Modernizar el canal de ventas y optimizar la gestión de inventario, pagos electrónicos (RF-03) y logística.
- **Requisitos Críticos Identificados:** La validación de requisitos realizada previamente enfatizó que el éxito del proyecto depende del cumplimiento de los RNF clave: **RNF-01 Usabilidad** (navegación intuitiva y *checkout* simple) y **RNF-07 Seguridad** (protección de datos y transacciones).

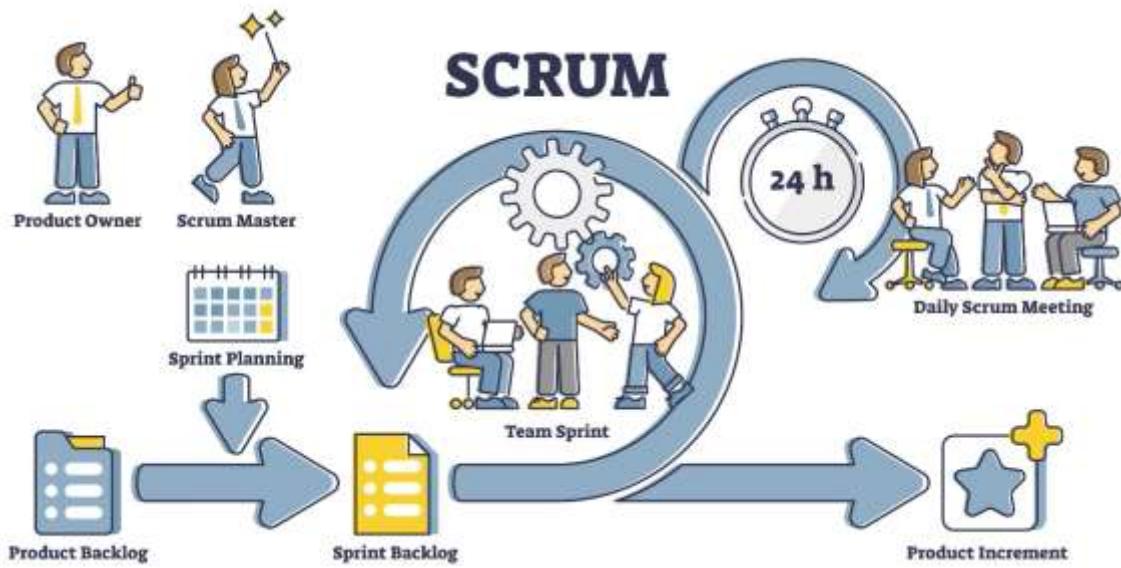
- **Entorno:** El proyecto opera en un entorno de **alta incertidumbre** y **expectativas dinámicas** del usuario final, lo que desaconseja cualquier enfoque de desarrollo lineal y rígido.

## 4. Marco Metodológico Seleccionado: Scrum

Scrum es un marco de trabajo que permite a equipos auto-organizados y multifuncionales entregar productos de alto valor de forma incremental e iterativa. Se basa en tres pilares: **Transparencia, Inspección y Adaptación**.

### Estructura del Ciclo de Vida (Gráfica Conceptual)

El desarrollo se organiza en ciclos fijos de tiempo de 2 semanas (*Sprints*), que generan un Incremento de Software potencialmente entregable.



## 5. Justificación Detallada de la Metodología

## 5.1. Contraste: Ágil (Scrum) vs. Tradicional (Cascada)

La siguiente tabla compara las características de Scrum frente a un modelo tradicional, enfocando las diferencias en el contexto del E-Commerce:

<b>Característica</b>	<b>Metodología Ágil (Scrum)</b>	<b>Metodología Tradicional (Cascada)</b>	<b>Impacto en Sport+ S.A.S.</b>
<b>Tolerancia al Cambio</b>	<b>Alta.</b> Los cambios se adoptan al inicio de cada <i>Sprint</i> mediante la re-priorización.	<b>Baja.</b> Los cambios tardíos paralizan el flujo y aumentan los costos.	El dinamismo de la industria deportiva (tendencias, promociones) exige una <b>respuesta inmediata</b> , que solo Scrum puede ofrecer.
<b>Pruebas y Calidad</b>	<b>Continuas.</b> Las pruebas se integran al final de cada <i>Sprint</i> ( <b>Definition of Done</b> ).	<b>Tardías.</b> Las pruebas se concentran al final del proyecto, detectando problemas de <i>core</i> demasiado tarde.	Permite validar la <b>Usabilidad (RNF-01)</b> y la <b>Seguridad (RNF-07)</b> en incrementos, evitando un costoso rediseño o corrección masiva.
<b>Entrega al Cliente</b>	<b>Frecuente.</b> Entregas de software funcional cada 2-3 semanas (MVP).	<b>Única y Tardía.</b> Entrega del producto funcional solo al final.	Genera <b>ROI (Retorno de Inversión)</b> más rápido y permite obtener <i>feedback</i> real del mercado para afinar la visión del producto.

## 5.2. Gestión de Riesgos y RNF Críticos

Scrum es seleccionado porque su estructura mitiga directamente los riesgos críticos del proyecto:

1. **Riesgo de Requisitos Ambiguos:** El **Daily Scrum** y la **Revisión del Sprint** fuerzan al equipo a inspeccionar el *software* funcionalmente, eliminando las ambigüedades que la documentación por sí sola no logra.
2. **Riesgo de Fallo en el RNF-07 (Seguridad):** Se incluirá el cumplimiento de las políticas de seguridad (ej. cifrado, validación de *inputs*) como un elemento obligatorio dentro de la **Definition of Done (DoD)** de cada *Sprint*. Si una *Historia de Usuario* no cumple con los criterios de seguridad, no se considera "Hecha".

## 6. Estructura de Roles, Artefactos y Herramientas

### 6.1. Roles Fundamentales

Rol	Responsabilidad Clave	En Sport+ S.A.S.
<b>Product Owner (PO)</b>	Maximiza el valor del producto gestionando el <i>Product Backlog</i> .	Representa la voz del negocio (Ventas y Marketing) y prioriza las <b>Épicas</b> (ej. "Flujo de Compra" o "Gestión de Inventario").
<b>Scrum Master (SM)</b>	Facilita el proceso, elimina impedimentos y protege al equipo de interrupciones externas.	Asegura que el equipo pueda enfocarse en la <b>excelencia técnica</b> , minimizando el tiempo perdido en problemas de coordinación.
<b>Equipo de Desarrollo</b>	Multifuncional, auto-organizado y responsable de la calidad y construcción de cada incremento.	Integrado por Analistas, Diseñadores UX/UI y Desarrolladores <i>Full-Stack</i> .

## 6.2. Artefactos de Gestión

Artefacto	Descripción
<b>Product Backlog</b>	Lista única, ordenada y priorizada de todo el trabajo conocido necesario para el producto (Funcionalidades y RNF).
<b>Sprint Backlog</b>	Conjunto de elementos del <i>Product Backlog</i> seleccionados para el <i>Sprint</i> actual, más el plan para entregar el Incremento.
<b>Incremento</b>	El resultado funcional y probado de un <i>Sprint</i> , que debe ser potencialmente entregable.

## 6.3. Herramientas de Apoyo (*Tooling*)

Para asegurar una gestión profesional, se utilizarán las siguientes herramientas:

- **Jira o Trello:** Para la gestión visual del *Product Backlog*, asignación de Historias de Usuario y seguimiento del progreso del *Sprint*.
- **Git / GitLab:** Como sistema de control de versiones y repositorio central de código, esencial para el trabajo colaborativo del equipo.
- **Integración Continua (CI/CD):** Se implementarán herramientas como Jenkins o GitLab CI para automatizar las pruebas unitarias y de integración, garantizando la calidad del código en cada *commit* (práctica de la **Programación Extrema - XP**).

## 7. Plan de Actividades Detallado bajo Scrum

Las fases de Diseño y Construcción serán ejecutadas a través de una serie de *Sprints* de dos semanas.

Fase	Sprint Sugerido	Objetivo Principal (Sprint Goal)	Actividades Clave de Diseño/Construcción
Diseño Lógico	Sprint 1	Definición del esqueleto técnico y lógico del sistema.	Elaboración del <b>Modelo Lógico de Datos (Diagrama de Clases)</b> . Definición de la arquitectura (Microservicios/Capas).
Diseño Dinámico	Sprint 2	Modelado de flujos de negocio y <i>setup</i> de la persistencia de datos.	Desarrollo de <b>Diagramas de Secuencia</b> para procesos críticos (Checkout y Login). Configuración del ORM y codificación inicial de la base de datos.
Construcción I (MVP)	Sprint 3	Implementación de las funcionalidades básicas que generan valor inmediato.	Construcción de las interfaces de <b>Registro/Login</b> y la vista de <b>Catálogo de Productos</b> . Implementación de pruebas unitarias.
Construcción II (Core)	Sprint 4	Implementación del corazón del e-commerce.	Implementación del <b>Carrito de Compras</b> (RF-02) y la lógica de actualización de stock (RF-05).

## 8. Conclusión

La adopción de **Scrum** es la decisión más estratégica para el desarrollo del E-Commerce Sport+ S.A.S. Su carácter empírico permite que el equipo se adapte a las necesidades del mercado con una eficiencia inalcanzable para los enfoques tradicionales. Al integrar prácticas de excelencia técnica (como TDD y CI/CD) dentro del marco de *Sprints*, se asegura que el producto no solo sea funcional, sino que también cumpla rigurosamente con los **RNF de Usabilidad, Rendimiento y Seguridad**, entregando una solución final robusta y sostenible.

## **9. Referencias Bibliográficas**

IEEE Computer Society. (2018). *ISO/IEC/IEEE 29148:2018 - Systems and software engineering - Life cycle processes - Requirements engineering*. IEEE.

Pressman, R. S. (2010). *Ingeniería del software: Un enfoque práctico* (7.<sup>a</sup> ed.). McGraw-Hill.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*. Scrum.org.

Servicio Nacional de Aprendizaje (SENA). (2024). *Fundamentos de análisis y diseño de software*. (Documento de apoyo para la formación).

Sommerville, I. (2011). *Ingeniería del software* (9.<sup>a</sup> ed.). Pearson Educación.