

**TRABAJO MONITOREO UNIVERSITARIO**

**Universidad Politécnica Internacional**

**Profesor**: SOFIA SOLANO CHACON

**Fecha de inicio:** 06 Noviembre 2024

**Fecha de conclusión:** 30 Noviembre 2024

**Carrera de estudio:** Ingeniería de Informática.

**Estudiante:**

Fabiola María Castro Mora

Diego Araya Porras

**Fecha:** 09/12/2024

[Fases de la metodología. 3](#_Toc341892128)

[1. Planificación del Sprint (Sprint Planning) 3](#_Toc56013413)

[2. Desarrollo del Sprint (Sprint Execution) 3](#_Toc1470437977)

[3. Revisión del Sprint (Sprint Review) 4](#_Toc2118632903)

[4. Retrospectiva del Sprint (Sprint Retrospective) 4](#_Toc3217160)

[5. Cierre del Sprint 4](#_Toc2143676358)

[Caso de Éxito: Spotify y la Implementación de SCRUM 5](#_Toc282906535)

[1. Técnicas de Estimación en Tiempo que Pudieron Aplicarse 5](#_Toc601271619)

[2. Herramientas de Planificación 6](#_Toc1711841337)

[3. Tiempo de Implementación 7](#_Toc378927643)

[4. Buenas Prácticas de Seguridad 7](#_Toc1378144787)

[5. Retos que Afrontaron 8](#_Toc436993160)

[Bibliografía 9](#_Toc1043433758)

1) Explicar los fundamentos y conceptos básicos de la metodología. (Randall GP)

2) Como funciona la metodología. (David Oconitrillo)

**Fases de la metodología.** (Fabiola Castro)

### **1. Planificación del Sprint (Sprint Planning)**

La fase de planificación marca el inicio de cada Sprint y se lleva a cabo al principio de este. En esta reunión, el equipo SCRUM, junto con el **Product Owner**, define qué se va a trabajar durante el Sprint y cómo se va a abordar.

**Actividades principales:**

* **Definir el objetivo del Sprint**: Se establece una meta clara que el equipo debe alcanzar al final del Sprint.
* **Selección de las historias de usuario o tareas**: El Product Owner presenta las historias del Product Backlog que se consideran prioritarias para el Sprint. El equipo de desarrollo selecciona las historias que podrán completar durante el Sprint.
* **Desglose de tareas**: El equipo desglosa las historias de usuario en tareas más pequeñas y manejables para facilitar su desarrollo.
* **Establecer un compromiso**: El equipo se compromete a completar las tareas seleccionadas durante el Sprint, teniendo en cuenta su capacidad y el tiempo disponible.

### **2. Desarrollo del Sprint (Sprint Execution)**

Esta fase abarca todo el trabajo que el equipo realiza durante el Sprint, que va desde la creación del código hasta las pruebas y la integración de las funcionalidades.

**Actividades principales:**

* **Desarrollo y pruebas**: Los desarrolladores crean las funcionalidades y realizan pruebas para asegurar que se cumplen los requisitos definidos en las historias de usuario. La calidad es una prioridad.
* **Daily Standup (Reunión diaria)**: Todos los días, el equipo realiza una reunión breve (de 15 minutos, aproximadamente) para sincronizarse. En esta reunión, cada miembro responde a tres preguntas:
  + ¿Qué hice ayer?
  + ¿Qué voy a hacer hoy?
  + ¿Hay algún obstáculo que me impida avanzar?
* **Trabajo colaborativo**: El equipo trabaja de manera colaborativa, ajustando el plan y resolviendo problemas en tiempo real para cumplir con los objetivos del Sprint.

### **3. Revisión del Sprint (Sprint Review)**

Al final de cada Sprint, se realiza una reunión de revisión para evaluar el trabajo completado y demostrar los avances al Product Owner y otras partes interesadas.

**Actividades principales:**

* **Demostración del trabajo completado**: El equipo presenta el incremento de producto desarrollado, mostrando las funcionalidades implementadas o los entregables.
* **Evaluación del Product Backlog**: Se revisa el Product Backlog para identificar los elementos que están listos para la siguiente iteración o si deben ajustarse según la retroalimentación recibida.
* **Feedback de los interesados**: El Product Owner, junto con otros stakeholders, brinda su retroalimentación sobre el incremento entregado, lo que puede resultar en ajustes al Backlog.

### **4. Retrospectiva del Sprint (Sprint Retrospective)**

Después de la revisión del Sprint, se lleva a cabo la reunión de retrospectiva. El objetivo de esta fase es reflexionar sobre el proceso y mejorar el desempeño del equipo en el siguiente Sprint.

**Actividades principales:**

* **Reflexión sobre el Sprint**: El equipo analiza cómo fue el Sprint desde un punto de vista del proceso, identificando lo que funcionó bien, lo que no, y qué se puede mejorar.
* **Identificación de áreas de mejora**: Se buscan soluciones para los problemas detectados en el Sprint anterior.
* **Definición de acciones**: Se acuerdan acciones específicas para mejorar el proceso en el siguiente Sprint, lo que puede implicar cambios en la forma de trabajar, herramientas o comunicación.

### **5. Cierre del Sprint**

Aunque no es una fase formalmente separada en la metodología SCRUM, el cierre del Sprint implica asegurar que todas las tareas y entregables se completaron según lo prometido. Después de la retrospectiva y la revisión, el ciclo se cierra y se inicia el siguiente Sprint.

4) Principios y valores. (David Oconitrillo)

5) Cuáles son los beneficios de la metodología. (Diego A)

Mejora de la colaboración y la comunicación: *Scrum promueve la colaboración y la comunicación a través de reuniones diarias, revisiones de sprint y retrospectivas que permiten al equipo compartir información y ajustar el trabajo de manera continua* (Schwaber & Sutherland, 2020).

Incremento en la productividad: *La naturaleza iterativa de Scrum permite a los equipos entregar incrementos de producto en plazos cortos, mejorando la productividad y proporcionando resultados valiosos de forma constante* (Rubin, 2013).

Flexibilidad y adaptabilidad: *Scrum permite a los equipos ser altamente adaptables a los cambios en los requisitos y en el entorno del proyecto, gracias a su enfoque en sprints cortos y revisión continua* (Sutherland & Schwaber, 2017).

6) Cuáles son las desventajas. (Diego A)

Dependencia de equipos altamente capacitados*: Scrum requiere un equipo con experiencia y habilidades para autoorganizarse y colaborar de manera efectiva, lo cual puede ser un desafío si los miembros no están familiarizados con la metodología o carecen de la capacitación necesaria* (Schwaber & Sutherland, 2020).

Complejidad en la implementación: *La implementación de Scrum puede resultar compleja en organizaciones que tienen una estructura jerárquica rígida, ya que el proceso implica un cambio cultural hacia un enfoque más ágil y colaborativo* (Sutherland & Schwaber, 2017).

Posible falta de documentación: *Dado que Scrum enfatiza el desarrollo de software funcional sobre la documentación extensa, puede haber casos en los que la falta de documentación formal cause problemas para la continuidad del proyecto o para nuevos miembros del equipo* (Rubin, 2013).

Riesgo de alcance descontrolado (scope creep*): El enfoque iterativo de Scrum, aunque permite la adaptabilidad, puede llevar al riesgo de un aumento descontrolado del alcance si no se gestiona adecuadamente* (Cohn, 2010).

7) Explicar un caso de éxito en que se haya implementado la metodología tomando en cuenta para la construcción de un cultura en la compañía y despliegue de sistemas, a considerar: (Fabiola Castro)

# Caso de Éxito: Spotify y la Implementación de SCRUM

Spotify es un caso paradigmático de cómo una empresa de tecnología puede usar la metodología SCRUM para transformar su cultura interna, mejorar su productividad y competir eficazmente en un mercado saturado. A continuación, se detalla cómo la implementación de SCRUM en Spotify le permitió fortalecer su estructura organizativa, optimizar su desarrollo de producto y mantenerse competitiva frente a gigantes como Google Music e iTunes.

#### 1. Técnicas de Estimación en Tiempo que Pudieron Aplicarse

Una de las primeras decisiones de Spotify al adoptar SCRUM fue implementar técnicas ágiles de estimación en tiempo, lo que permitió a los equipos planificar y entregar valor de manera eficiente. Las principales técnicas que podrían haberse aplicado son:

* Puntos de Historia: Spotify, al igual que muchas empresas ágiles, probablemente utilizó la estimación en puntos de historia para evaluar el tamaño relativo de las tareas o historias de usuario. Esto les permitió determinar el esfuerzo necesario para completar ciertas funcionalidades, sin caer en la trampa de estimaciones demasiado precisas basadas en horas.
* Planning Poker: Una de las técnicas ágiles más comunes para estimar las historias de usuario es el Planning Poker, donde los miembros del equipo asignan puntos de historia a cada tarea en función de su complejidad. Esto se hace de manera colaborativa, asegurando que todos los miembros del equipo tengan voz y visión en la planificación.
* Velocidad (Velocity): Una vez que los equipos establecieron una base de estimaciones, usaron la velocidad como una métrica clave para medir el rendimiento del equipo. La velocidad se refiere al número de puntos de historia completados durante un Sprint. Con esta métrica, Spotify pudo prever con mayor precisión cuántos puntos podrían completar en futuros Sprints y planificar mejor los lanzamientos de características.

#### 2. Herramientas de Planificación

Spotify adoptó varias herramientas ágiles de planificación y gestión para coordinar sus esfuerzos en múltiples equipos y Sprints:

* JIRA: Spotify usó JIRA, una de las herramientas más populares para la gestión de proyectos ágiles. JIRA permitió gestionar el Product Backlog, el Sprint Backlog y las tareas de los equipos en tiempo real. Además, la herramienta integraba la estimación en puntos de historia y proporcionaba métricas sobre el rendimiento de los equipos.
* Confluence: Para la documentación colaborativa y el seguimiento de las decisiones tomadas en los Sprints, Spotify utilizó Confluence. Esta herramienta les permitió centralizar la información sobre el desarrollo de la plataforma, incluyendo los diseños, requisitos y análisis del cliente, que son fundamentales para mantener la transparencia y la alineación en los equipos.
* Slack: Como herramienta de comunicación interna, Spotify utilizó Slack para coordinar a los equipos de forma rápida y efectiva. Slack facilitó la comunicación diaria, sobre todo en equipos distribuidos, permitiendo que la información fluyera sin barreras entre los miembros del equipo y los Scrum Masters.
* Miro o Trello: Algunas áreas más creativas dentro de Spotify pudieron haber utilizado herramientas visuales como Miro o Trello para la planificación de flujos de trabajo y procesos de diseño, lo cual es común en equipos ágiles que priorizan la colaboración y la creatividad.

#### 3. Tiempo de Implementación

La implementación de SCRUM en Spotify no fue un proceso inmediato, sino un proceso gradual de transformación. Se puede dividir en fases:

* Fase 1: Formación y Introducción a SCRUM (0-3 meses): Spotify comenzó con la formación de sus equipos en SCRUM y la contratación de expertos en Agile, lo que permitió a la empresa comprender las ventajas del marco de trabajo. Se implementaron los roles clave de SCRUM, como Scrum Masters y Product Owners, y se capacitó a los equipos para que pudieran adoptar las prácticas ágiles.
* Fase 2: Piloto y Adopción Inicial (3-6 meses): Durante los primeros meses, Spotify implementó SCRUM en equipos pequeños como prueba piloto. Los equipos trabajaron en Sprints de 2 semanas para gestionar el desarrollo de funcionalidades específicas dentro de la plataforma. Fue un proceso de adaptación, y se ajustaron las herramientas y procesos según las necesidades de cada equipo.
* Fase 3: Escalado de SCRUM a Toda la Empresa (6-12 meses): Después de la fase piloto, Spotify empezó a escalar SCRUM a toda la organización. Esto significó que todos los equipos (desarrollo, diseño, marketing, etc.) comenzaron a trabajar bajo el mismo marco de SCRUM, lo que llevó a una integración más fluida y una mayor colaboración entre diferentes departamentos.

#### 4. Buenas Prácticas de Seguridad

Aunque Spotify es principalmente una plataforma de música y no se enfrenta a las mismas amenazas de seguridad que una empresa financiera, la seguridad sigue siendo una prioridad para ellos. Algunas de las buenas prácticas de seguridad que podrían haber aplicado son:

* Integración de Seguridad en el Pipeline de Desarrollo: Spotify, como muchas empresas ágiles, habría integrado las pruebas de seguridad directamente en el proceso de desarrollo, asegurándose de que las vulnerabilidades fueran identificadas y corregidas antes de que el software se lanzara a producción.
* Pruebas Automatizadas de Seguridad: Se habría empleado análisis estático de código y pruebas automatizadas de penetración para garantizar que los sistemas fueran seguros, especialmente cuando se integran nuevas funcionalidades o cambios en la plataforma.
* Revisión de Código en Pares: La revisión de código en pares (peer review) es una práctica común en equipos ágiles para asegurar la calidad del código y detectar posibles vulnerabilidades de seguridad antes de la implementación.
* Autenticación y Autorización Segura: Dado que Spotify maneja grandes volúmenes de datos de usuario, asegurarse de que la plataforma cumpla con los estándares de autenticación (como OAuth) y autorización (como permisos de acceso) es esencial para garantizar la protección de la información personal.

#### 5. Retos que Afrontaron

La implementación de SCRUM en Spotify no estuvo exenta de desafíos. Los principales retos que enfrentaron fueron:

* Resistencia al Cambio: Como en cualquier organización, al principio hubo resistencia al cambio. Algunos empleados y gerentes acostumbrados a procesos más tradicionales no comprendían los beneficios de trabajar de forma ágil. Spotify enfrentó este reto con capacitación y sesiones de coaching para explicar los principios de SCRUM y cómo mejoraría la eficiencia general.
* Escalabilidad de SCRUM: Dado que Spotify creció rápidamente, la escalabilidad del marco SCRUM se convirtió en un desafío. Con equipos distribuidos geográficamente y proyectos cada vez más complejos, mantener la coherencia entre los equipos y coordinar múltiples Sprints fue complicado. Para resolver esto, introdujeron el concepto de Squads, que son equipos ágiles multidisciplinarios que trabajan de forma autónoma pero alineados a través de objetivos comunes. También implementaron la figura de Tribes para coordinar varios Squads relacionados.
* Falta de Sincronización entre Equipos: Con el crecimiento de los equipos, algunos de los objetivos de las tareas se desalinearon, y la sincronización entre los diferentes equipos se volvió más difícil. Spotify abordó este desafío con reuniones de Scrum de Scrums, donde Scrum Masters de diferentes equipos se reúnen para compartir avances, retos y dependencias.
* Gestión del Cambio Cultural: A medida que Spotify adoptaba SCRUM, la cultura organizacional también debía transformarse. Era necesario cambiar la mentalidad de los empleados hacia un enfoque más autónomo y colaborativo. Para lograr esto, Spotify promovió una cultura de confianza, transparencia y apertura, donde los equipos se sentían empoderados para tomar decisiones sin la necesidad de aprobación constante.

Spotify es un ejemplo claro de cómo SCRUM no solo mejora la gestión de proyectos y el desarrollo de software, sino que también puede transformar la cultura organizacional. A través de la adopción de SCRUM, Spotify logró mejorar la productividad, aumentar la colaboración entre equipos y fomentar un entorno de trabajo más ágil y autónomo. Este enfoque permitió a la empresa mantenerse competitiva frente a gigantes de la música digital, mejorar su plataforma y ofrecer un producto que satisface continuamente las demandas de sus usuarios, todo mientras mantenía un ambiente de trabajo motivador y eficiente.

8) Conclusiones de la metodología investigada. (Diego A)

9) Si usted tuviera que implementar la metodología para la construcción de su sistema que acciones diferentes al caso práctico investigado haría. (Randall GP)

3. RECOMENDACIONES DE LECCIONES APRENDIDAS COMO CONSULTOR (2%) 4. REFERENCIAS (obligatorio) Debe incluir aquellas fuentes de información que consultó y deberán ser fuentes confiables, sitios web, libros, revistas, entre otros. Debe utilizar la versión 7 de APA. (Randall GP) (David Oconitrillo)

# Bibliografía

Cohn, M. (2010). *Succeeding with Agile: Software Development Using Scrum.* Addison-Wesley.

Mancuzo, G. (2023, 22 noviembre). ▷ 5 fases de la metodología SCRum 【5 etapas y 16 procesos】. *ComparaSoftware*. <https://blog.comparasoftware.com/fases-metodologia-scrum/>

Ortega, E. H. (2021, 22 septiembre). *Scrum y sus casos de éxito*. CLEFormación - Formación TIC Para Profesiónales. <https://cleformacion.com/scrum-y-sus-casos-de-exito/>

Rubin, K. S. (2013). *Essential Scrum: A Practical Guide to the Most Popular Agile Process.* Addison-Wesley.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org.

Sutherland, J., & Schwaber, K. (2017). *Scrum: The Art of Doing Twice the Work in Half the Time*. Crown Business.+