

**Semana 03- Audrey Gerardo Juárez.**

## **¿Cómo implementar SCRUM?**

La lectura se divide en 11 pasos a seguir, es una guía para poder empezar de manera correcta con la metodología Scrum:

El paso número 1 es elegir a un “responsable del producto”, quién será una persona visionaria y activa que se comprometa con el proyecto. El punto 2 y 3 son igualmente de importantes y son muy parecidos al primero, ya que, nos pide seleccionar al personal, sólo que está vez a los que conformaran nuestro equipo, al igual que al Scrum Máster, respectivamente.

El punto 4 y 5 se refieren a la “bitácora de producto”, se nos explica que es una lista de alto nivel de todo lo que debe hacerse para volver realidad la visión. En los proyectos existe sólo una, las decisiones que se marquen ahí se deben priorizar y consultarse con todo el equipo para cerciorar el hecho de que se tenga una representación correcta.

Más adelante nos comentan que es importante medir el esfuerzo que se tomara al realizar las tareas, el equipo debe guiarse con preguntas clave refiriéndose a tiempos, descripciones, información importante, valor del proyecto, entre otras.

En la idea número 6 nos informan sobre una de las bases más importantes de Scrum, que es la “planeación”, pero sobre todo la “planeación de Sprints”, estos deben de ser de extensión fija e inferiores a un mes.

Se nos habla de igual manera sobre la importancia del aspecto visual en el proyecto, refiriéndose específicamente al “tablero Scrum” donde se colocan 3 columnas “Pendiente, En proceso y Terminado”. Dentro de las columnas se colocarán notas adhesivas con los pendientes del proyecto en curso.

Diariamente se deben efectuar unas juntas de no más de 15 min de duración, dónde se nos planteen cuestionamientos de a cuerdo al desempeño del equipo y el cómo mejorar la eficiencia y el apoyo grupal.

La “revisión, demostración y posterior retrospectiva” de un Sprint, son importantes, se dedica el punto 9 y 10 a este tema. Se nos dice que todos pueden asistir, es una reunión abierta del equipo en la que se muestra la culminación de sus operaciones. Pero lo más importante es comprender que el éxito del trabajo del equipo, debe ser documentada, así como la mejora continua que debe ser constante hasta en esta última fase del proyecto. También se nos habla sobre que muchas veces puede ser no tan grato recibir la crítica, pero debemos tomarlo con madurez y comprender que es constructiva y que eso sólo nos convierte en seres más capaces a la hora de efectuar nuestro trabajo.

Finalmente comienza de nuevo el ciclo, según nos lo plantea el punto 11.

# SPRING BATCH

## 1. ¿Qué es Spring Batch?

Es un “lightweight y framework para facilitar el desarrollo de aplicaciones por lotes sólidas. También proporciona servicios técnicos más avanzados que admiten trabajos por lotes de alto rendimiento y volumen extremadamente alto a través de sus técnicas de optimización y partición.

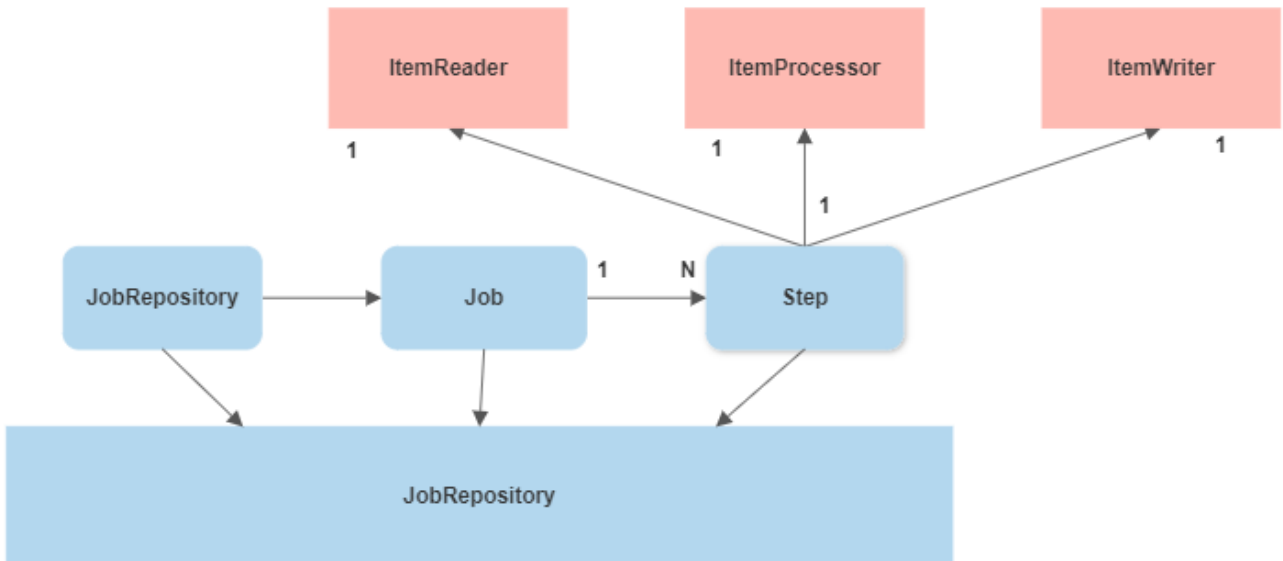
## 2. ¿Qué resuelve y en dónde se usa?

- ✓ Procesamiento automatizado y complejo de grandes volúmenes de información que se procesa de manera más eficiente sin la interacción del usuario.
- ✓ Aplicación periódica de reglas comerciales complejas procesadas repetitivamente en conjuntos de datos muy grandes (por ejemplo, determinación de beneficios de seguros o ajustes de tarifas).
- ✓ Integración de información que se recibe de sistemas internos y externos que generalmente requiere formateo, validación y procesamiento de manera transaccional en el sistema de registro.

## 3. Alternativas a Spring Batch (en Java):

- **Hadoop.** Maneja de igual manera, grandes volúmenes de datos.
- **Talend Open Studio For Data Integration.** Es una plataforma de código abierto basada en Java que pretende ser un all-in-one del manejo de datos.
- **Spring Boot.** Un marco basado en Java de código abierto que es posible utilizar para crear un microservicio.
- **Apache Spark.** Es un motor de análisis unificado para el procesamiento de datos a gran escala
- **Kafka.** Crea canalizaciones de datos de transmisión en tiempo real y aplicaciones que se adapten a los flujos de datos.

## DIAGRAMA SPRING BATCH



### Explicación diagrama:

**Job** es a la tarea que se va a crear y cada tarea/job, puede estar compuesta por 1 o varios **Step** (pasos), a su vez cada Step/paso tiene las actividades de “lectura (**ItemReader**), procesamiento (**ItemProcessor**) y escritura (**ItemWriter**)”.

**JobRepository** comunica spring batch con la base de datos, ver en tablas el estado de los job's que se encuentren en ejecución. A su vez **JobLauncher** invoca a la ejecución de este ciclo.