



Gustavo A Arellano

@arellano_gus

Programmer, Thinker also known as
Goose by friends and mates.

📍 México DF

🔗 gustavo-arellano.com

💻 Se unió en julio de 2010

🖼 184 fotos y videos

Introducción a un popular proceso de desarrollo ágil: Scrum

M. en C. Gustavo A. Arellano

©2022

@arellano_gus

Disclaimer

- El presente seminario es presentado en una modalidad de “OpenKnowledge” que confiere el derecho a reproducirlo y modificarlo libre de todo cargo, a cualquier persona siempre y cuando ésta respete y conserve:
 - a. Esta nota (licencia)
 - b. La información de autoría del documento y
 - c. La re transmisión también sea libre de todo cargo

Reseña Gustavo Arellano

- Matemático, Facultad de Ciencias, UNAM
- Maestro en Ciencias, especialidad en Ingeniería de Software
- Java Certs:
 - SCJP, SCJD, SCWCD, SCEA
- Scrum Certs:
 - SCF, SCD, SPOC, SCM, AEC
- Security Certs ECCouncil:
 - Ethical Hacker, Secure Programmer, Forensic Investigator
- Apple Software Engineerier
- GE Project leader (Schennectady, Atlanta, Stamford, CT)
- Experiencia real en industrias de USA, Chile, Ecuador & España
- 1 artículo escrito en el área de Inteligencia Artificial:
 - Antelope (Redes regulatorias de Genes)
- DG CONACYT, DGA en la CNBV, DG en la CNS, CTO en Infotec
- Académico en el IIMAS y en la Facultad de Ciencias de la UNAM
- **Senior Software Architect @ Bitso**

Generales

- Scrum es una metodología de trabajo en equipo que sirve como guía para la adecuada gestión de diversos tipos de proyectos, **incluidos los proyectos de desarrollo de Software**
- Es particularmente popular en el ámbito de la Ingeniería de Software



Gustavo A Arellano @arellano_gus · 10 ene.

Scrum es un conjunto de procesos, prácticas, actividades, roles y artefactos que interactúan de manera armónica para lograr articular una mecánica ágil, colaborativa, iterativa, incremental, integral y autorganizada de trabajo cuyo fin es alcanzar una meta común. Qué Opinan?

Características fundamentales

- Algunas de las características mas relevantes de Scrum son:
 - Generación de productos con un numero de defectos menor que el que se produce con otras metodologias
 - Maximiza las posibilidades de entregar productos con alto apego a los requerimientos del cliente
 - Reduce la inversión de esfuerzo invertido en el control y la administración de equipos para su ajuste a las normativas del proceso
 - Promueve el fortalecimiento de equipos auto dirigidos y auto organizados. Equipos independientes y autosuficientes que requieren de muy poca guía.
 - Conduce a la madurez de equipos altamente motivados, lo que a su vez, incrementa su creatividad y sus propuestas de innovación.

Bajo numero de defectos

- Scrum establece diversos mecanismos de control que aseguran la reducción de defectos que normalmente serían injectados al usar otros modelo metodológico. Algunos de estos mecanismos son:
 - Mecanismo de “Pull Request”
 - Esfuerzos cortos, enfocados e intensos de trabajo
 - Reunión de revisión con el propietarios del producto al final del ciclo (sprint)
 - Generación obligatoria de Pruebas Unitarias y factor de Cobertura mínima alto.
 - Inclusión de mecanismos automáticos de detección de omisiones, violaciones y/o apego a estándares
 - Transparencia en el manejo de la información y cualquier problemática derivada del día a día de trabajo

Apego a los criterios del cliente

- Scrum define los llamados “Acceptance Criteria” como parte medular de su proceso de captura de requerimientos.
- Como tomador de decisiones, un CIO puede estar particularmente interesado en adoptar un modelo que integre esta actividad, ya que con ella se garantiza la inmutabilidad de un requerimiento y es, al mismo tiempo, una forma de asegurar que lo solicitado es lo presentado.

Reducción de los controles administrativos

- Dado que Scrum establece cinco tipos de reuniones de control, un administrador global requiere de menos inversión de tiempo en la supervisión del equipo. La reuniones de Scrum cubren la totalidad del espectro de desarrollo:
 - Daily standup meeting
 - Grooming
 - Planning
 - Review
 - Retrospective

Equipos auto suficientes

- Las siguientes son características de un equipo Scrum:
 - Cross funcionales
 - Auto dirigidos
 - Auto organizados
- De lo anterior, un equipo Scrum es altamente productivo y mantiene una motivación que les da autonomía e independencia.
- Este tipo de equipo es mucho mas económico que el equipo que requiere de continua capacitación de terceros y continuo monitoreo y/o seguimiento

Estructura General

- Scrum se encuentra estructurado de la siguiente manera:
 - Sprints
 - RSP
 - Roles
 - SM, PO, Team
 - Ceremonias
 - Daily, Grooming, Planning, Review, Retrospective
 - Acuerdos, Valores y Principios
 - DoD, transparencia, colaboración, etc.
 - Conceptos generales y artefactos (entregables e internos)
 - **User Story**, Epicas, PBL, Acceptance Criteria, etc. (ver sig. diap.)

Artefactos (entregables e internos)

1. Historias de Usuario
2. Product backlog
3. Sprint backlog
4. Release Schedule Plan (RSP)
5. Notas de retrospectivas y los “Impediment Logs”
6. Cálculos de planeación (auto asignaciones, story points, cargas, etc)
7. Casos de Uso
8. Maqueta del sistema (Mock., WireFrame)
9. Matriz de pruebas, datos de prueba y evidencia de su ejecución
10. Gráfica de Burndown
11. Actas de las ceremonias (menos standup) –Minutas-
12. Evidencia de las pruebas de estrés y concurrencia
13. Esquema de base de datos y diagramas de E/R
14. Diagramas de diseño

User Story



As a <role>
I want <goal>
so that <benefit>

Acceptance criteria:
(Conditions of Satisfaction)

...

...

As an Account Manager
I want a sales report of my account
to be sent to my inbox daily
So that I can monitor the sales
progress of my customer portfolio

Acceptance criteria:

1. The report is sent daily to my inbox
2. The report contains the following sales details: ...
3. The report is in csv format.

Product Owner

- Definir la visión del proyecto.
- Ayudar a finalizar la elección del **Scrum Master** para el proyecto.
- Identificar a los stakeholders.
- Ayudar a determinar a los miembros del equipo Scrum.
- Crear épicas y prototipos.
- Priorizar los elementos de la lista priorizada de pendientes del producto ([Product Backlog](#)).
- Definir los criterios de terminado.
- Ayudar a crear historias de usuario.
- Explicar las historias de usuario al equipo Scrum, al tiempo que crea la lista de tareas.
- Mantener la lista priorizada de pendientes del producto.
- Aceptar/Rechazar los entregables.
- Representar a los usuarios del producto o servicio con un profundo conocimiento de la comunidad de usuarios.
- Asegurar los recursos financieros del proyecto.

Scrum Master

- Ayudar a identificar a los stakeholders para el proyecto.
- Facilita la selección del equipo Scrum.
- Garantizar que los recursos de respaldo estén disponibles para el funcionamiento del proyecto sin problemas.
- Ayudar al **Product Owner** en la creación de la lista priorizada de pendientes del producto y en la definición de criterios de aceptación.
- Determina la duración del sprint.
- Apoya al equipo Scrum en la estimación del esfuerzo necesario para completar las tareas acordadas para el sprint
- Ayuda a actualizar el tablero Scrum y el registro de impedimentos.
- Facilita las reuniones de revisión de la lista priorizada de pendientes del producto.
- Se asegura que los problemas que afectan al equipo Scrum se discutan y se resuelvan.
- Garantizar que exista un ambiente ideal para el equipo Scrum.

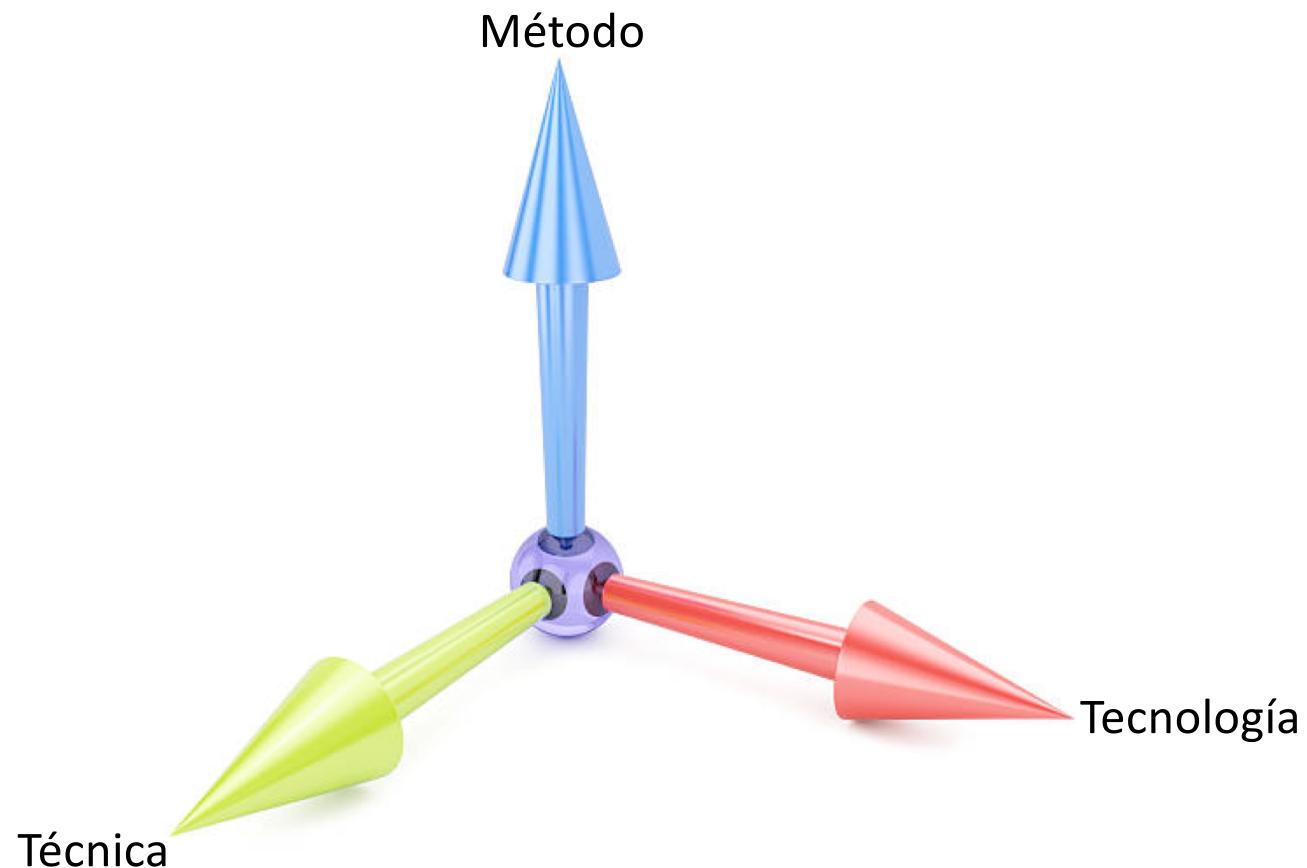
Team

- Asegura una comprensión clara de las épicas y prototipos.
- Entender las historias de usuario en la lista priorizada de pendientes del producto.
- Estar de acuerdo con los demás miembros del **Scrum Core Team** sobre la duración del Sprint.
- Estimar las historias de usuario aprobadas por el **Product Owner**.
- Asignar las historias de usuario que se hacen en un Sprint.
- Desarrollar la lista de tareas en base a las historias de usuario ya convenidas y las dependencias.
- Crear entregables.
- Actualizar el registro de impedimentos y las dependencias.
- Actualizar la tabla del trabajo pendiente y el tablero Scrum.
- Realizar las reuniones diarias de pie (**Daily Standup Meeting**).
- Identificar oportunidades de mejora en la reunión de retrospectiva del sprint.
- Participar en la reunión de retrospectiva del proyecto.

Instrumentando Scrum

- Toda metodología debe ser independiente de cualquier herramienta
- Una buena herramienta siempre incrementará los niveles de productividad durante la ejecución de cualquier proceso de desarrollo
- Por ejemplo, algunas empresas han tomado la decisión de adoptar la suite de productos Atlassian, para instrumentar su proceso de desarrollo, aunque usen Scrum incluso para proyectos que no son de Software

Ejes rectores



Metodología

- La guía sistemática y organizada como un conjunto de roles, actividades y sus artefactos resultantes. Ejemplo:
 - Unified Process (<https://www.programaenlinea.net/proceso-unificado-rational-rup/>)
 - Extreme programming (Ken Becket) (Peer review, Unit Test, Refactoring)
 - **Scrum** ([manifiesto Agil](#))
 - SAFe (Scrum Escalado)
- Lo anterior se encuentra frecuentemente ligado con la disciplina de gestión de proyectos y control de avance (PMP).
- De igual manera, tópicos como el costeo, el cálculo de ruta crítica, la gestión de riesgo, etc. son temas con un fuerte enfoque metodológico.

Tecnología

- El conjunto de herramientas, entre las que se encuentran los lenguajes, los frameworks, los IDE's, etc. Ejemplo:
 - Java
 - GIT
 - Spring
 - React
 - Jenkins
 - Nexus
 - Sonar
 - Go
 - Docker
 - MariaDB

Técnica

- La correcta forma de hacer algo o usar algo.
Ejemplos:
 - Uso de una llave de cruz
 - Uso de un rotomartillo
 - Generación de una Arquitectura
 - Empleo de un patrón
 - Generación de un diseño
 - Selección de un paradigma (OOP, AOP, etc)
 - Uso de una técnica (TDD, BDD, DDD, etc)
 - Adopción de una práctica (GitFlow)

RUP

Core Process Workflows

Business Modeling

Requirements

Analysis & Design

Implementation

Test

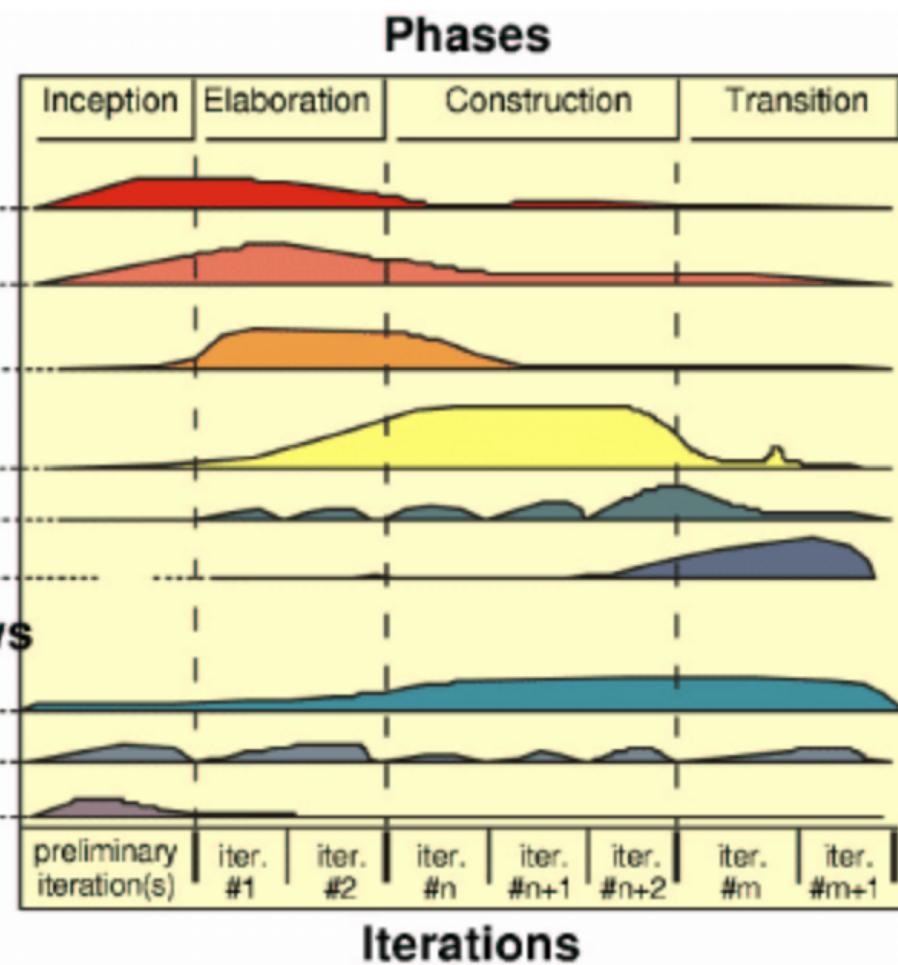
Deployment

Core Supporting Workflows

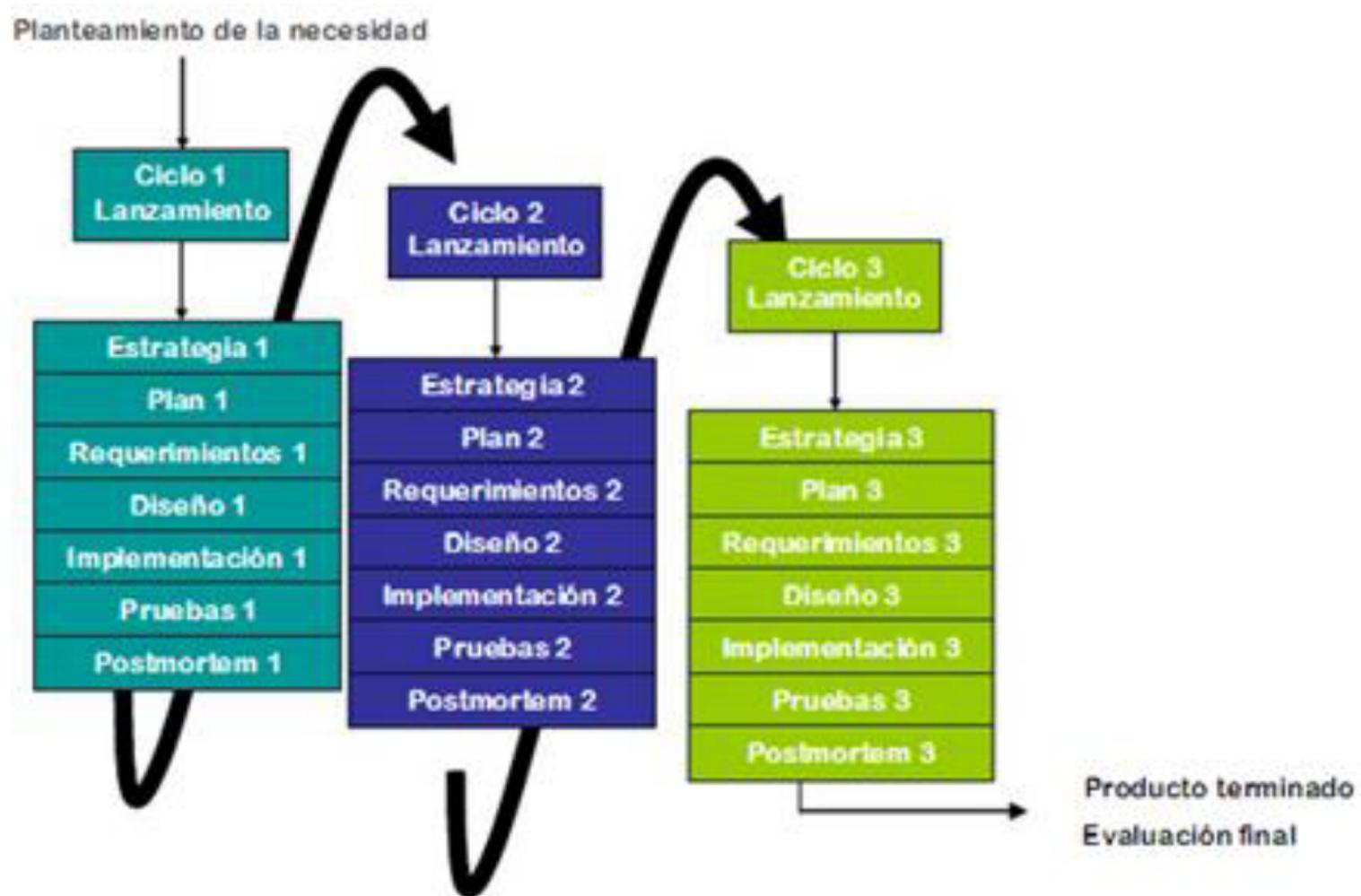
Configuration & Change Mgmt

Project Management

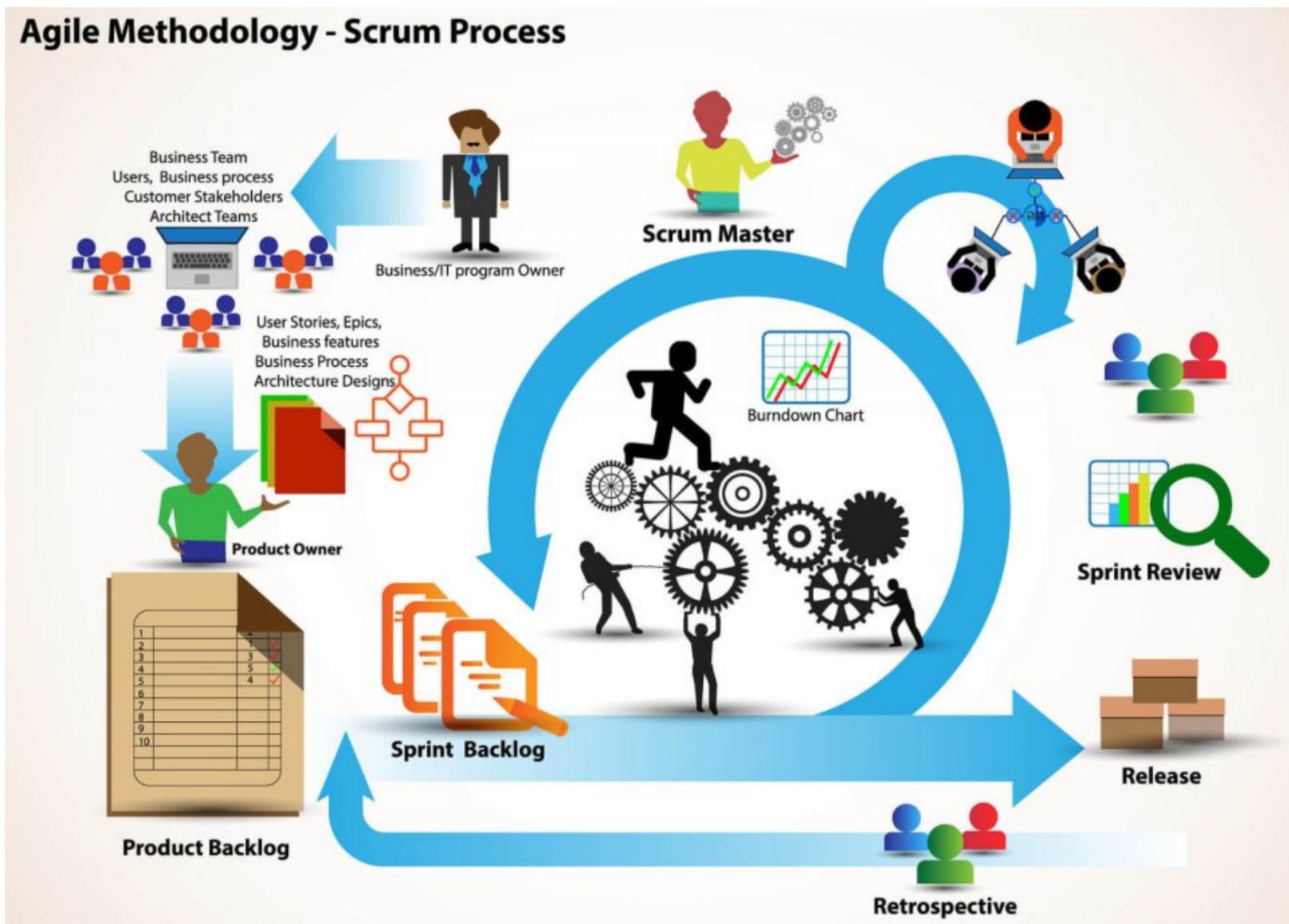
Environment



T S P



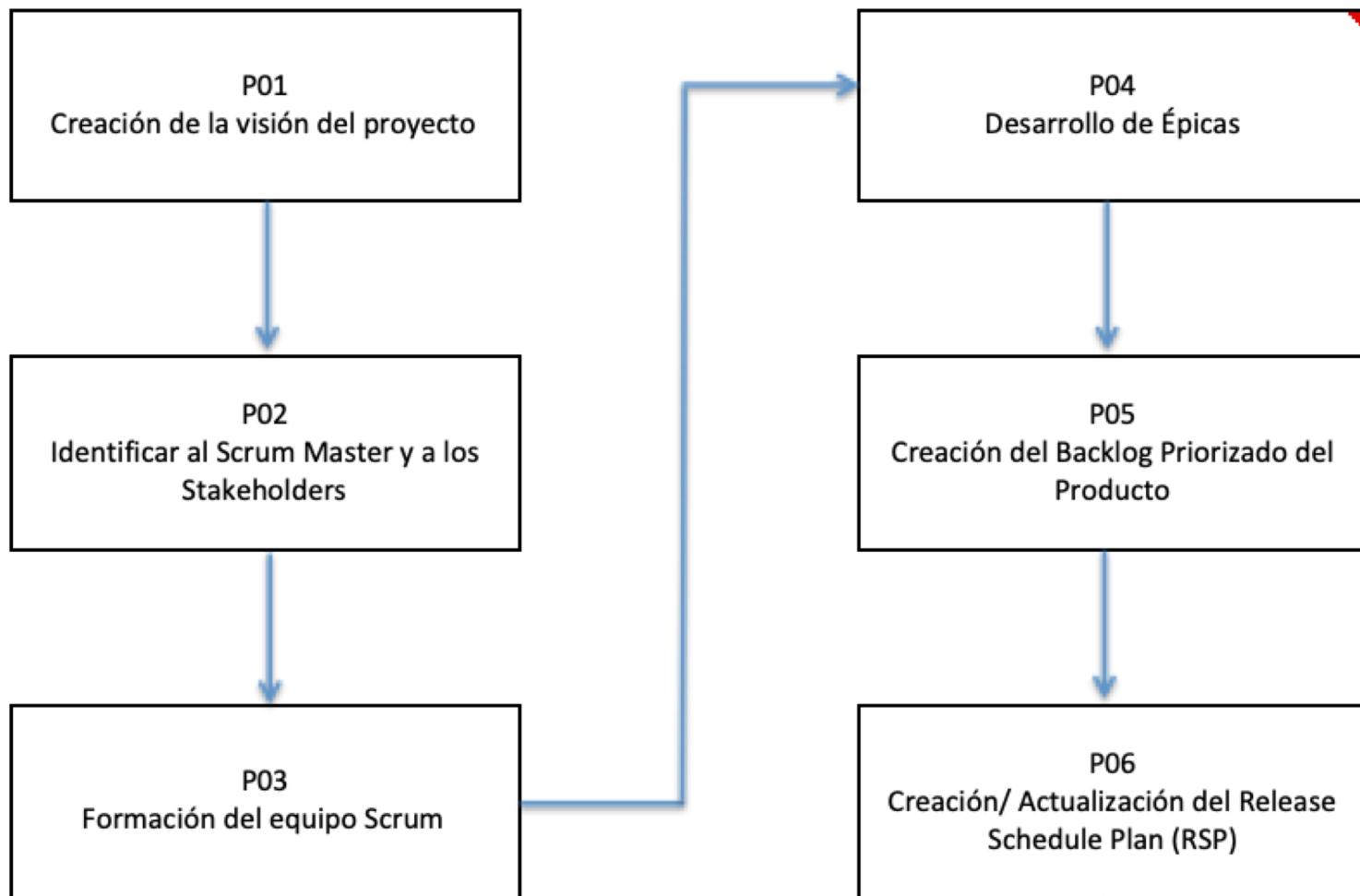
Agile Methodology - Scrum Process



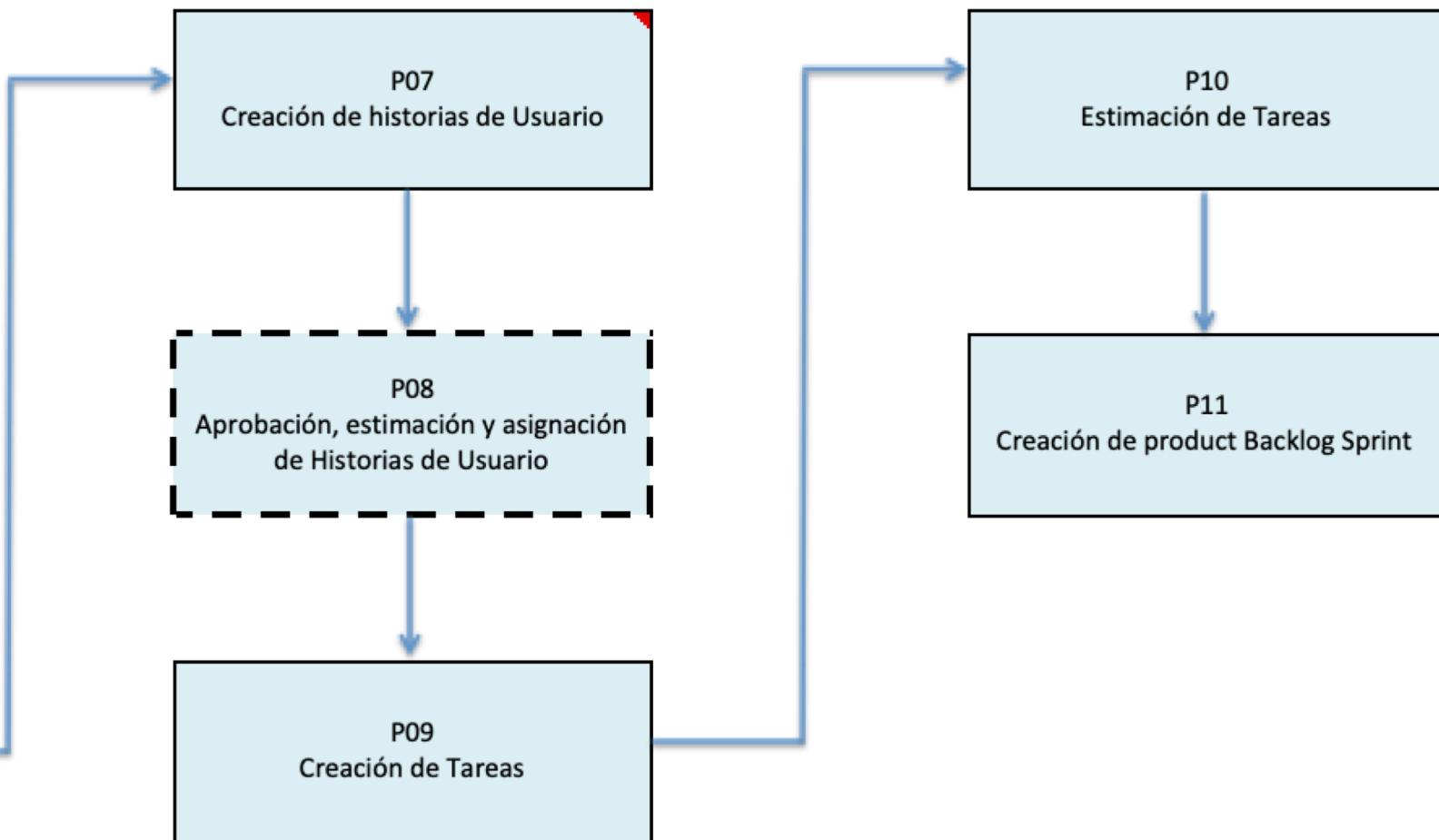
Los 17 sub procesos de Scrum

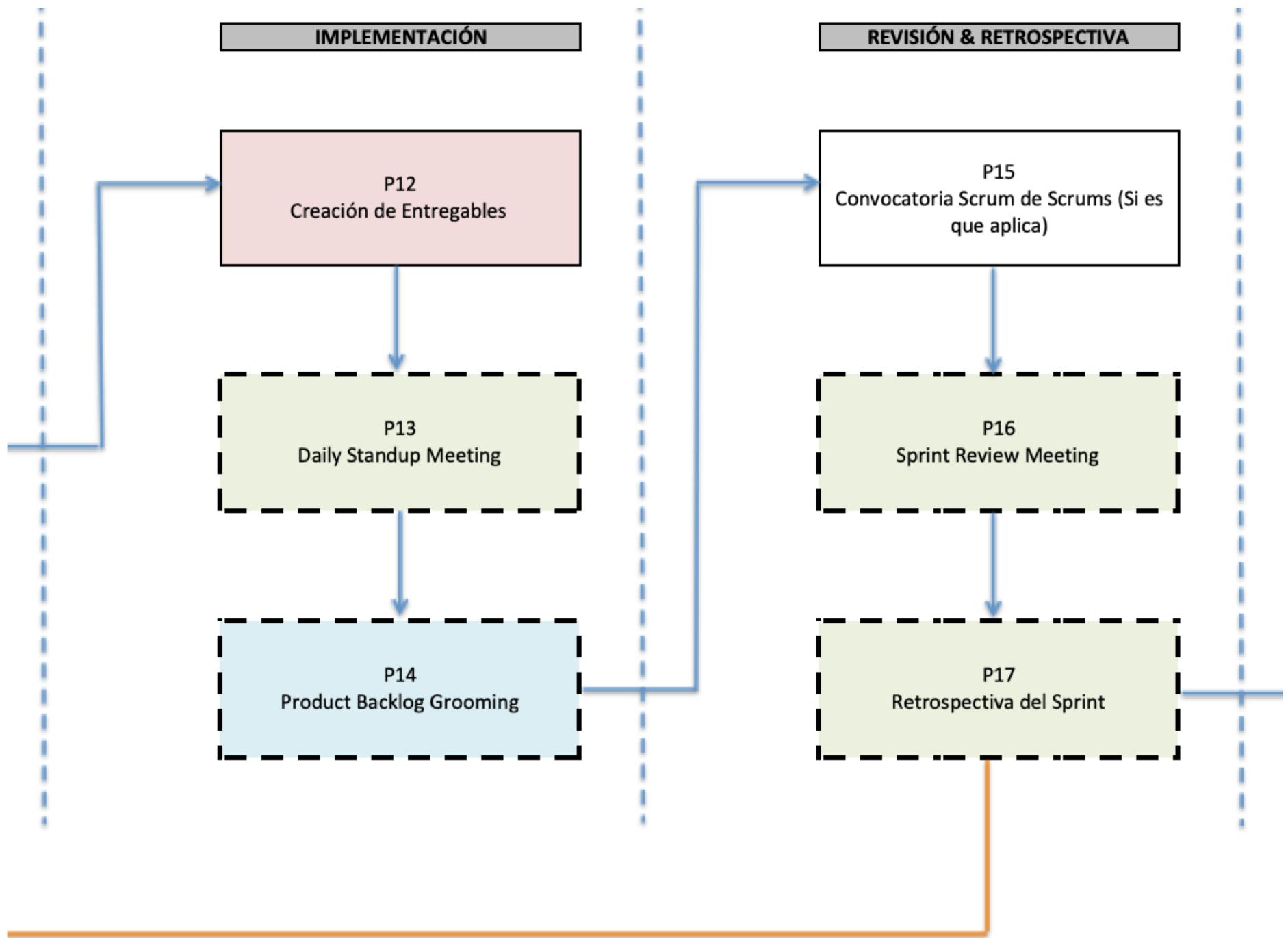
- La metodología SCRUM está organizada en 17 sub procesos que conservan una secuencia de ejecución específica.
- Algunos sub procesos son ejecutados únicamente al inicio del proyecto, algunos únicamente al final y algunos son ejecutados de forma iterativa o recurrente.
- A continuación se explica de manera general cada uno de ellos.

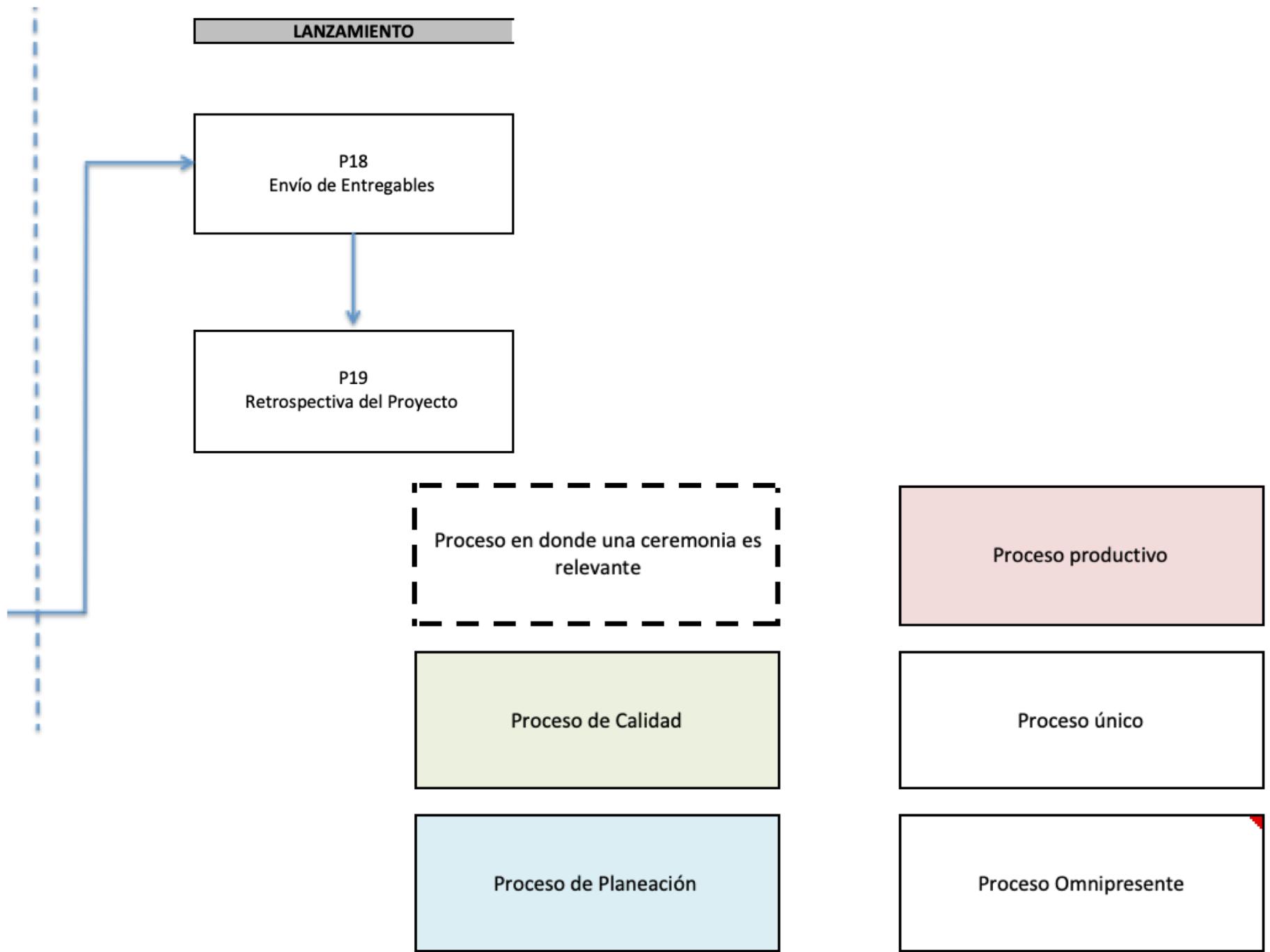
INICIO

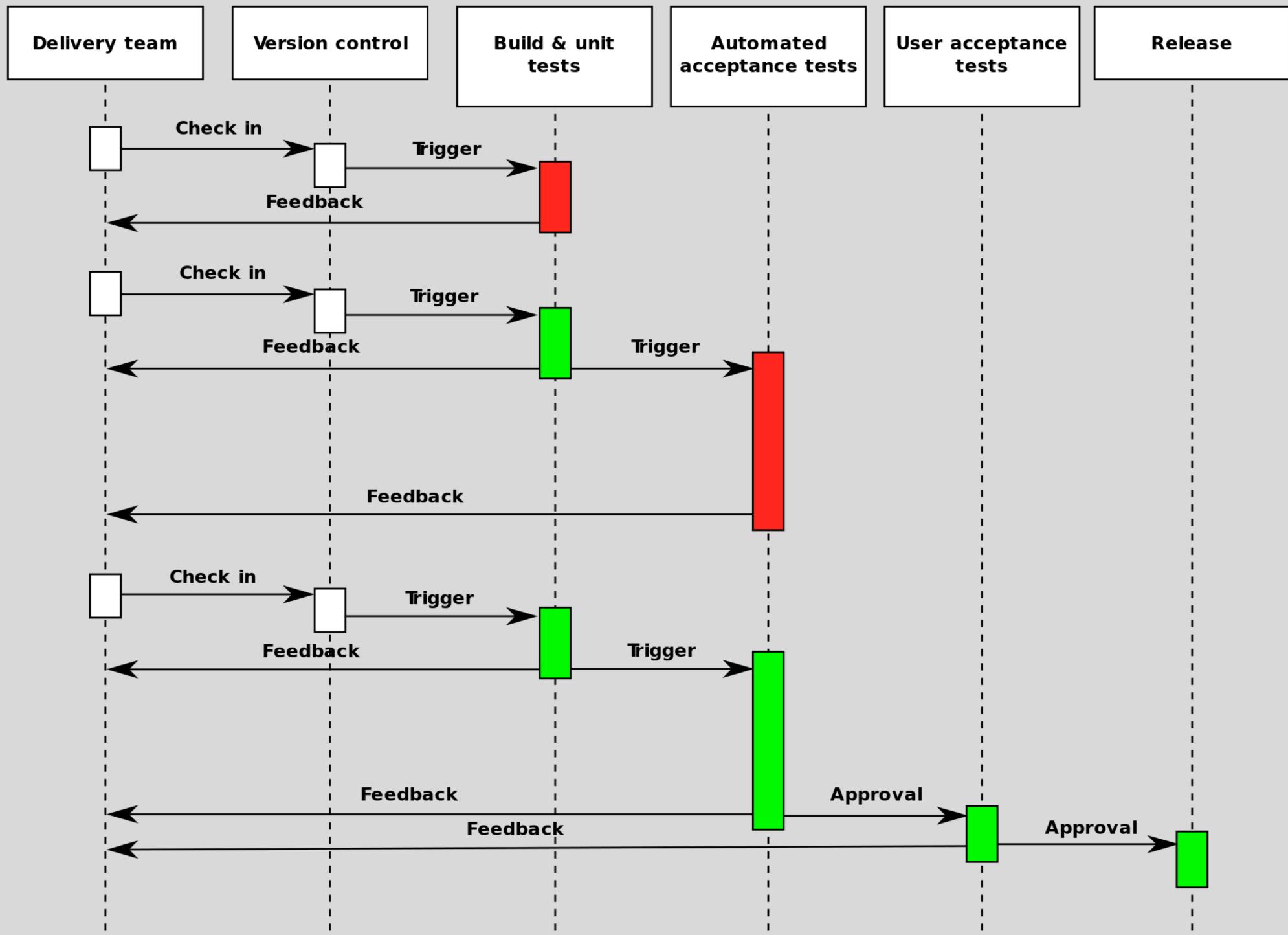


PLANEACIÓN & ESTIMACIÓN









Herramientas de soporte para incremento en la productividad

- Según nuestra experiencia, las herramientas que a continuación se mencionan, han contribuido significativamente a incrementar la productividad en nuestros desarrollos:
 - Confluence → Gitlab
 - JIRA → Gitlab
 - Stash → Bitbucket → GitLab
 - Bamboo → Jenkins
 - Infrared → Docker (que es, mas bien, una tecnología)

JIRA → Bitbucket → eclipse

Apache
maven



spring
boot



spring

Nexus



git

Atlassian
Bamboo

ubuntu



amazon
web services™



ANSIBLE

MariaDB

docker

RANCHER

{...} swagger

Gracias

G. Arellano

2022

@arellano_gus