

Administración de proyectos

Repasso

Objetivos de la administración de proyectos

- Entregar el software en el tiempo acordado
- Mantener el presupuesto
- Cumplir con las funcionalidades solicitadas
- Mantener un equipo de desarrollo óptimo



Actividades del administrador de proyectos

1. Planeación del proyecto
2. Informes
3. Gestión de riesgo
4. Gestión de personal
5. Creación de propuestas





Retos de los proyectos de software

- El producto del trabajo suele ser intangible
- Proyectos grandes pueden ser inmanejables
 - Tecnologías cambiantes
 - Mucha experiencia en lógica de negocio
- Los proyectos son sumamente cambiantes
- Los proyectos se adaptan a la organización
- Los proyectos suelen ser innovadores para una organización.



Gestión de
Personal

“

*El personal es el activo más
importante de una empresa*



Administrador de proyecto y la gestión del personal

- La organización debe obtener el mejor rendimiento posible de su inversión en personal.
- Se debe estar al tanto de problemas potenciales en el personal.



Conocimiento vs. Habilidades humanas

- Los ingenieros de software tienen el conocimiento técnico para saber qué se necesita.
- Los ingenieros de software tienen la capacidad de entrenar al personal.
- Los ingenieros de software con frecuencia carecen de habilidades para motivar a un equipo.
- Los ingenieros de software con frecuencia carecen de habilidades para dirigir.
- Los ingenieros de software muchas veces no pueden delegar funciones.
- Los ingenieros de software no siempre pueden comunicar problemas de manera asertiva.

Responsabilidades del administrador de proyectos de software:

- La carga de trabajo va acorde a las habilidades y experiencia del empleado
- Respeto a los trabajadores y sus intereses
- Se mantiene motivado al empleado
 - Reconocimiento de su esfuerzo
 - Se le dan oportunidades de crecimiento
 - Se le dan oportunidades de desarrollo personal



Factores críticos

- Consistencia
- Respeto
- Inclusión
- Honestidad





Motivación del personal

Un colaborador motivado contribuirá con lo mejor de sus habilidades.

Trabajadores motivados!

- Motivación significa organizar el trabajo y el ambiente laboral para alentar a los individuos a desempeñarse tan efectivamente como sea posible.
- Si las personas no están motivadas, no estarán interesadas en la actividad que realizan.



Necesidades de autorrealización para la motivación

Necesidades de autorrealización
Necesidades de estima
Necesidades sociales
Necesidades de seguridad
Necesidades fisiológicas



Maslow 1954.



Identificar personalidades del equipo

- Personas orientada a tareas
- Personas orientada hacia sí misma
- Personas orientadas a la interacción



Modelo de
madurez de
gestión de personal

Modelo de Madurez de Gestión de Personal

- “Aumentar la rapidez con la cual las organizaciones de software acometen las aplicaciones cada vez más complejas al ayudar a atraer, aumentar, motivar, desplegar y retener el talento necesario para mejorar su capacidad de desarrollo de software”

CUR94

URL: <https://www.redalyc.org/pdf/4259/425942453008.pdf>

ayuda a las organizaciones a mejorar sus procesos de gestión de personal, proporciona un marco para atraer, desarrollar , motivas, organizar y retener el talento necesario para mejorar continuamente y entregar proyectos de calidad.

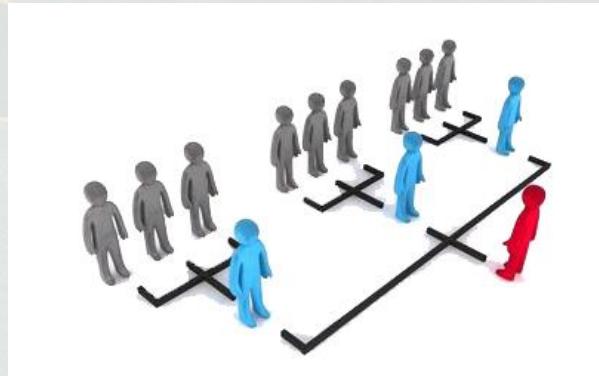
Modelo de Madurez de Gestión de Personal

- La necesidad de contar con personal para el desarrollo del software altamente preparado y motivado se viene discutiendo desde los años 60
- El MMCGP es compañero del modelo de madurez de la capacidad software, que guía a las organizaciones en la creación de un proceso de software maduro.

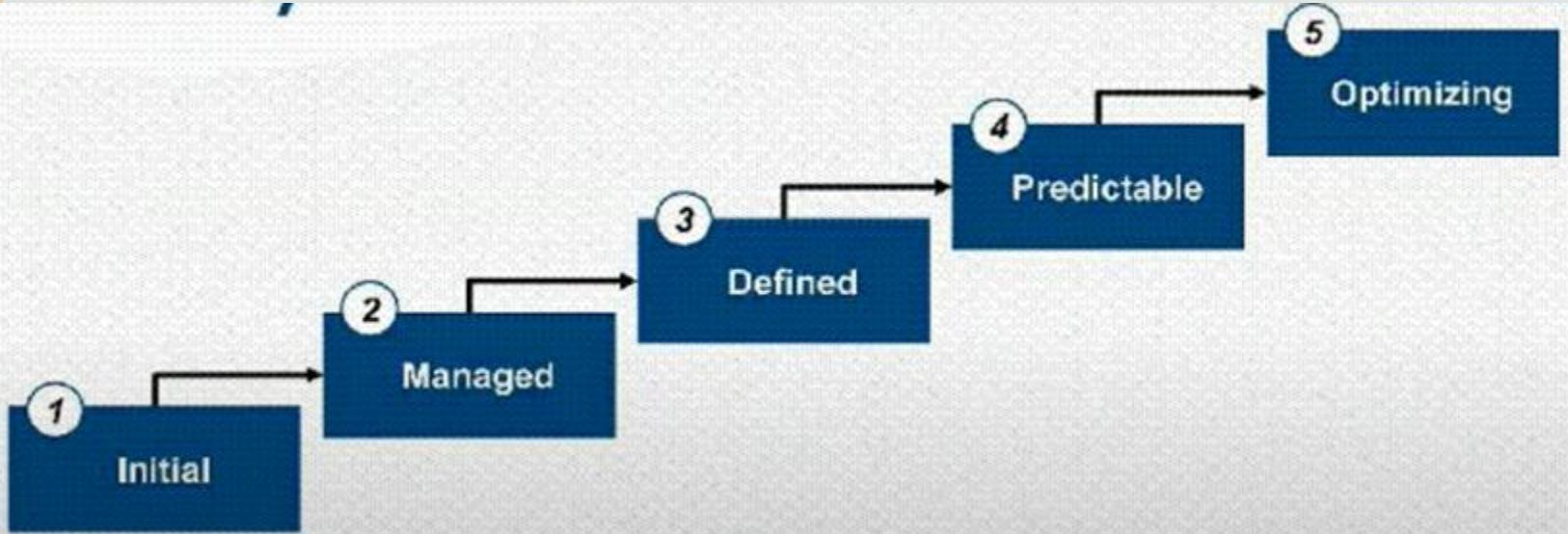


Áreas Clave

- Reclutamiento
- Selección
- Gestión de rendimiento
- Entrenamiento
- Retribución
- Desarrollo de carrera
- Diseño de organización
- Desarrollo de la cultura de equipo



Modelo de Madurez de Gestión de Personal



Reclutamiento y Selección

- Buscar el equilibrio de habilidades técnicas y personalidades.
- Es importante que todos puedan trabajar en conjunto.
- Es importante que un nuevo miembro esté dispuesto a aprender.
- Intentar ver más allá.

Proceso de selección

- Evaluar parte técnica.
- Evaluar personalidad.
- Evaluar compatibilidad en el equipo.
- Evaluar habilidades blandas.



Entrenamiento

- Cómo funciona el negocio?
- Procesos internos de la empresa
- Procesos tecnológicos
- Buenas prácticas
- Nuevas tecnologías
- Nuevas atribuciones



Gestión de Rendimiento

- Rae: Producto o utilidad que rinde o da alguien o algo.
- Conocer la eficiencia de producción por factor y recurso empleado brindará información útil sobre la gestión de recursos en la empresa.
- Necesidad de acciones correctivas
- Reconocer el buen trabajo

La gestión de rendimiento es el proceso a través del cual una organización involucra a su personal para mejorar la eficacia en la consecución de los objetivos de la empresa. Este proceso incluye la definición de expectativas, la evaluación del rendimiento del empleado, y el uso de esa evaluación para motivar, desarrollar y mejorar el desempeño futuro. Es un enfoque continuo y colaborativo entre el empleado y el supervisor, enfocado en el logro de metas y el desarrollo profesional.



Factores que impactan el rendimiento del personal

- Factores técnicos
- Factores de la organización
- Factores Productivos
- Factores Personales
- Factores de Liderazgo



Calculando el rendimiento

- Evaluación de desempeño
 - Habilidades
 - Cualidades
 - Necesidades en el equipo

Intentar hacer objetivo algo que es muy subjetivo



A Definir

- Criterios de medición = punto de comparación
- Indicadores = aspectos/acciones de productividad a medir
- Establecer un sistema de ponderación
- Definir período de tiempo

Métodos para medir el rendimiento

- Rendimiento laboral = (Productos o Servicios Producidos) / (Recursos Utilizados)

A collage of mathematical diagrams and formulas:

- Geometry: A right triangle with vertices A and B, hypotenuse AB, and legs x_1 , y_1 . The equation $y = a(x-b)2+c$ is shown.
- Volume: A sphere with radius r and volume $V = \frac{4}{3}\pi r^3$.
- Trigonometry: A circle with radius r and angle $\cos(\frac{\pi}{6}) = \frac{\sqrt{3}}{2}$.
- Algebra: The equation $(\frac{a}{b})^c = \frac{a^c}{b^c}$ is shown.
- Angles: A diagram showing angles $a+b+c+d=360^\circ$ and $a+b+c=180^\circ$.
- Pythagorean Theorem: $a^2 + b^2 = c^2$.
- Logarithms: $\log_a 1 = 0$ and $\log_a(\frac{a}{b}) = \log_a a - \log_a b$.
- Series: The formula $\sum_{k=0}^n k = \frac{1}{2}n(n+1)$.
- Approximation: $\pi \approx 3.14$.
- Trigonometric Functions: $\sin 30^\circ = \frac{1}{2}$, $\sin 45^\circ = \frac{1}{\sqrt{2}}$, $\sin 60^\circ = \frac{\sqrt{3}}{2}$.
- Volume: A rectangular prism with dimensions l, w, h and surface area $SA = 2lw + 2lh + 2wh$.
- Trigonometric Identity: $\sin^2 y + \cos^2 y = 1$.
- Circle Properties: A circle with radius r and circumference $2\pi r$.
- Algebraic Identities: $a^3 + b^3 = (a+b)(a^2 - ab + b^2)$ and $(8^3)^3 = 8^{3 \times 3} = 8^9$.
- Trigonometric Functions: A graph of $y = \sin x$ from $-\pi$ to 2π .
- Pythagorean Theorem: $c^2 = a^2 + b^2$.
- Trigonometric Ratios: $\tan 60^\circ = \sqrt{3}$.
- Area: The formula $A = sr$ where $r = \frac{A}{s}$.
- Trigonometric Functions: A right triangle with hypotenuse $\sqrt{2}$ and angle 45° .
- Area: The formula $C = 2\pi r$.

Ejemplo

- Supongamos que la compañía fabrica 40 botellas de whisky cada día, con un valor de 80 euros por cada una. El empleado en cuestión labora 4 horas al día, percibiendo una tarifa de 15 euros por hora, y embotella 4 botellas diarias.
- La fórmula para evaluar el rendimiento laboral del empleado es la siguiente: Rendimiento laboral del empleado = (Cantidad de productos que el empleado produce x Precio unitario del producto) / (Horas trabajadas diariamente x Tarifa horaria).

Rendimiento vs Productividad

- **Rendimiento:** obtener resultados deseables para una compañía
- **Productividad:** acciones que se llevan a cabo para obtener resultados

Rendimiento se refiere a qué tan eficazmente un empleado o equipo cumple con sus tareas específicas, considerando la calidad y los resultados obtenidos.

Productividad, en cambio, mide la eficiencia de la producción, es decir, la cantidad de salida producida en relación con los recursos utilizados, como el tiempo o el material.

Mientras el rendimiento se enfoca más en la calidad y el cumplimiento de los objetivos, la productividad hace énfasis en la cantidad y el uso eficiente de los recursos.

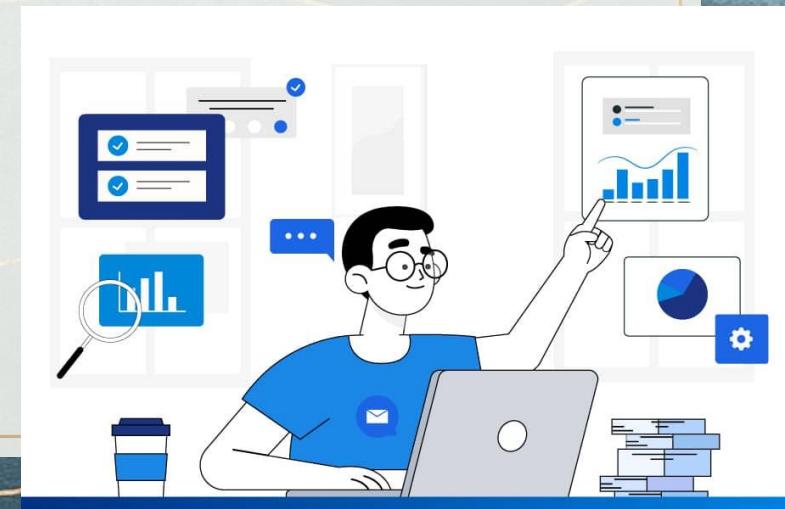
Para el software...

- Se requiere algo de intuición y otros factores
- Habilidades blandas
- Trabajo en equipo
- Capacidad de adaptarse
- Capacidades técnicas
- Lenguaje de programación específico
- Creatividad



Algunos métodos

- Método de la puntuación
- Descripción
- Reunión
- Autoevaluación
- Evaluación del compañero
- Evaluación 360
- Retroalimentación del cliente
- Encuestas
- Híbrido



Otro Ejemplo

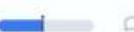
	Media esperada	Equipo de diseño	Carmen Torres	Consuelo Nava...	David Rodríguez	Jimena Menén...	Laura Ruiz
Soft Skills	4.6	3.4	3.2	4.6	2.9	2.8	2.9
Trabajo en equipo	4.1	4.0	2.4	4.7	2.8	4.5	2.9
Capacidad de adaptación	3.2	0.0	2.1	4.3	1.9	4.9	3.1
Technical Skills	3.5	4.7	4.8	4.5	1.0	3.2	4.1

Otro ejemplo

Evaluado
Maria Del Pilar Quintanilla Pérez
Marketing - Técnico - Granada - Director de Marketing.

Evaluador principal
Antonio Flores Quevado:
Peso: 100%
Nota final: 75 %
Finalizado

Valoración final
Balance (Gap)
+33% 
Ha obtenido un 29% de lo que se podía recibir

Competencias	Tipo	Requerido	Obtenido	Balance
Automotivación	Competencias Generales	Nivel 2 - 3	3 - Óptimo	
Trabajo en equipo	Competencias Generales	Nivel 3 - 3	2 - Regular	
Calidad	Competencias Generales	Nivel 2 - 4	3 - Alto	
Adaptación al cambio	Competencias Generales	Nivel 3 - 3	3 - Óptimo	
Lealtad	Valores	Nivel 3 - 4	3 - Se podría mejorar	
Integridad	Valores	Nivel 3 - 4	4 - Óptimo	
Compromiso con la empresa	Valores	Nivel 3 - 3	2 - Regular	
Comunicación fluida	Empresa y cliente	Nivel 2 - 3	3 - Óptimo	
Orientación al cliente	Empresa y cliente	Nivel 2 - 3	3 - Óptimo	

Acciones Correctivas

- No todo está perdido
- Entrenamientos de ser necesarios
- Llamada de atención
- Acompañamiento sobre algún tema en específico
- Ajustes en la actividad o proyecto
- Crear conciencia



Retribución

- Bono de productividad
- Aumento
- Promoción
- Libertades dentro del equipo
- Convenio entre la empresa y el empleado

Desarrollo de Carrera

- Los miembros del equipo se vuelven más y más expertos.
- La tecnología es cambiante
- Es más y más fácil adquirir conocimiento
- Las generaciones actuales buscan superarse
- Es un tipo de retribución
- Es conveniente para la empresa



Diseño de organización

- Deben Existir procesos bien definidos
- Los miembros del equipo deben contar con el equipo necesario para hacer su trabajo
- Los miembros del equipo deben contar con información necesaria para realizar su trabajo
- Deben tener claras las asignaciones laborales





Trabajo en Equipo

Trabajo en Equipo

- Un grupo demasiado grande no puede trabajar en equipo.
- **Trabajar en grupo:** los miembros son independientes entre sí y tienen responsabilidades individuales.
- **Trabajo en equipo:** los miembros colaboran en conjunto con sinergia para resolver problemas.

El trabajo en grupo implica la realización de tareas de manera colectiva, donde cada miembro contribuye de forma más o menos independiente, sin que necesariamente exista una interdependencia o un objetivo común claro.

En cambio, el trabajo en equipo se caracteriza por una fuerte interdependencia entre sus miembros, quienes comparten un objetivo común y colaboran estrechamente, coordinando sus esfuerzos y apoyándose mutuamente para alcanzar dicho objetivo. La sinergia y la cohesión son claves en el trabajo en equipo.

Trabajo en equipo

- En general, un equipo no debería exceder las 10 personas.
- Deberían estudiar el proyecto de forma grupal.
- Todos deberían reunirse y tener lluvia de ideas.
- Todos deberían aportar en la estimación de tiempos.
- Todos deberían participar en la planificación.
- Si surge un problema, todos deberían ser parte de la solución.
- Todos deberían compartir el conocimiento adquirido

Desarrollo de la cultura de equipo

- Se deben promover actividades de integración de equipos.
- Se debe promover que los miembros del equipo socialicen para que se lleven bien.
- Se debe reconocer el esfuerzo en conjunto.
- Se debe facilitar la comunicación entre los miembros del equipo.
- Reconocer la importancia de cada rol en el equipo.



Importancia de la cultura de equipo

- Confianza entre los miembros del equipo
- Distribuir adecuadamente las habilidades de cada miembro para aplicarlas a un problema
- El conocimiento debe compartirse





Liderazgo

Líderes de equipo

- Motivación: Producir a la mayor capacidad posible
- Organización: adecuar procesos o inventar nuevos
- Ideas o Innovación: habilidad para alentar a las personas a crear aún cuando deben trabajar bajo límites establecidos.

“

*Un líder es aquel que sabe a
dónde quiere ir, se levanta y
va...*

John Erskine

Gestor de proyectos – Rasgos Clave

- Resolución de problemas
- Dotes de Gestión
- Incentivar
- Influencia y fomento de la cultura de equipo



Paradigmas organizacionales

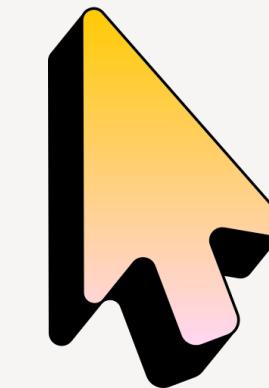
- **Paradigma Cerrado:** jerarquía tradicional de autoridad, trabajan mejor cuando producen software similar al anterior.
- **Paradigma Aleatorio:** depende de la iniciativa individual de los miembros del equipo.
- **Paradigma Abierto:** involucra mucha innovación a la hora de resolver problemas. Es un trabajo sumamente colaborativo.
- **Paradigma sincrónico:** se comparte un problema y se organiza a los miembros para trabajar en partes del problema con poca comunicación entre ellos.

Integración y Entrega continua

AGENDA

Desarrollo ágil

Reutilización del software



Entregas continuas

Integración continua

Desarrollo ágil



Desarrollo ágil

- Entorno Cambiante
- Agilizar el proceso de desarrollo



Métodos ágiles

- Desarrollo Incremental
- Normalmente al final de una iteración hay una liberación en un entorno productivo.



CI - Continuous Integration

Integración de código

Integración de Cambios

Divide y vencerás

Verificación de cambios

Trabajo en Equipo

Colaboración sin prepotencia

Estrategia del cambio

Trabajo a distancia



CI/CD es una metodología de desarrollo de software que combina la Integración Continua (CI) con la Entrega o Despliegue Continuo (CD).

La CI se refiere a la práctica de fusionar automáticamente y con frecuencia el trabajo de desarrollo en un repositorio común, donde se ejecutan pruebas automáticas para detectar errores lo antes posible.

El CD extiende este proceso al automatizar la entrega del código a entornos de producción o de prueba, facilitando despliegues rápidos y seguros. Juntos, CI/CD permiten un desarrollo más eficiente, rápido y de alta calidad.

CD - Continuous Delivery

Dividir Entregas en algo más pequeño

El proceso de instalación se realiza 1 vez

Enfoque en tareas específicas

Ahorro de tiempo

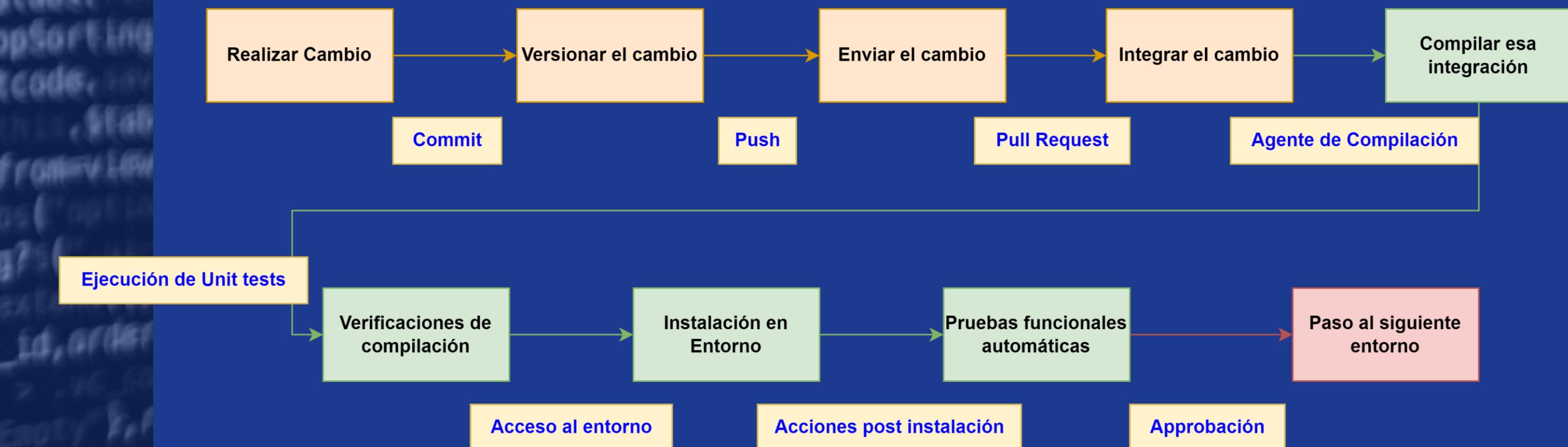
Verificaciones automáticas

Trabajo a Distancia

Agrega valor al cliente



Tareas en el proceso:



Otras Tareas / Políticas

Compilar código enviado

Code Coverage

Publicar actualización utilizando un servicio interno ya en un entorno

Aprobación de pruebas unitarias

Pruebas funcionales automáticas

Ejecución de scripts de actualización de data

Backups antes de instalar



(CODECOVERAGE) es una medida que indica qué porcentaje del código fuente de un programa ha sido ejecutado por las pruebas automatizadas. En otras palabras, nos dice qué parte del código se ha verificado para que funcione correctamente.

Es importante tener una buena cobertura de código porque nos ayuda a asegurarnos de que nuestro programa está bien probado y es menos probable que tenga errores. También nos ayuda a identificar las partes del código que no se han probado, lo que nos permite enfocarnos en escribir más pruebas para esas áreas.

Beneficios del proceso definido:

- Reducción del riesgo
- Roles bien definidos
- Rollback en caso de emergencia
- Ahorro de tiempo
- Evitar “Re-Work”
- Trabajo en paralelo

Reparaciones en el camino

Tomar la versión correcta

Realizar los cambios o reparaciones pertinentes

Integrar los cambios

Compilar

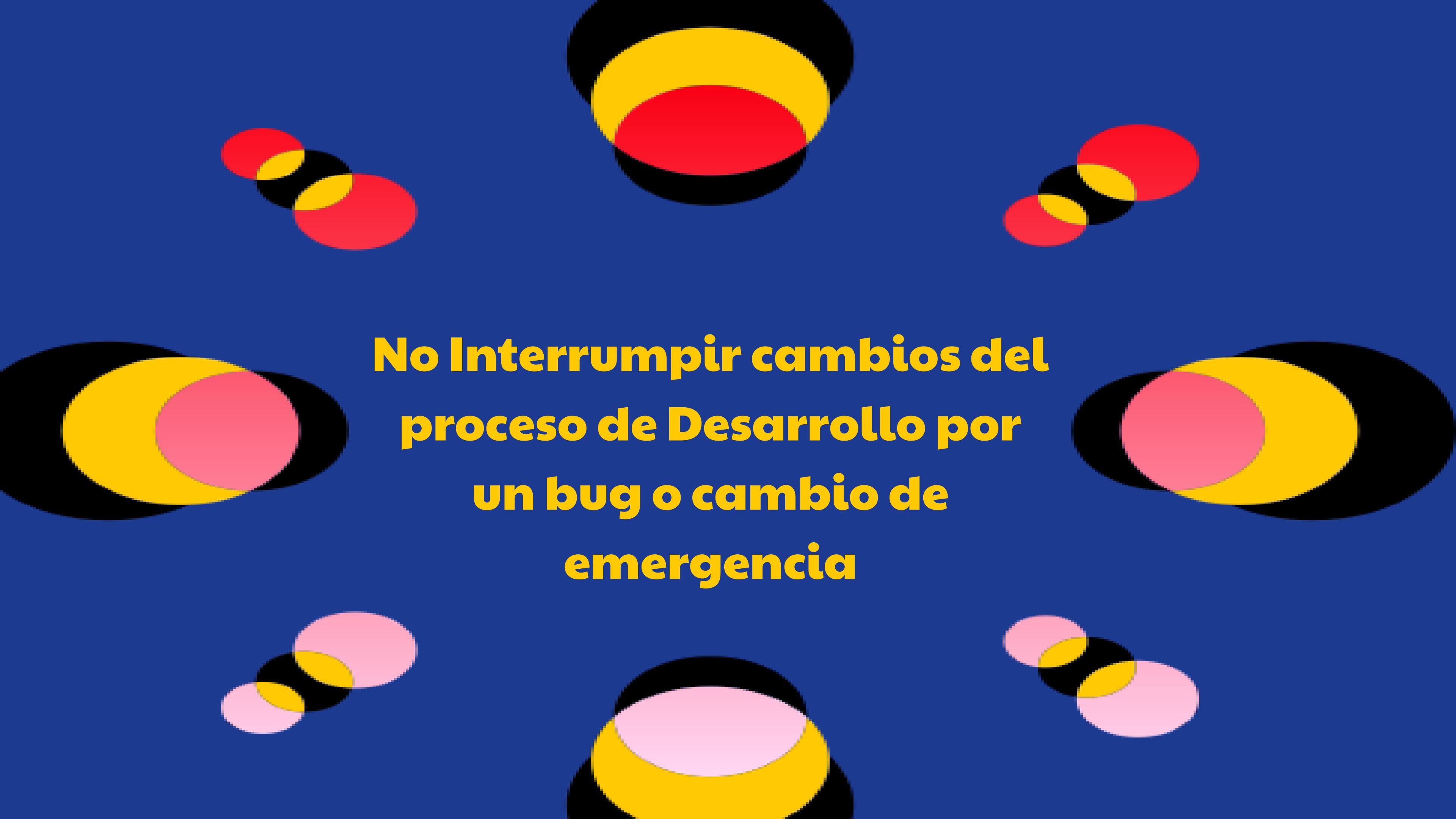
Instalar en entorno de pruebas

Verificar

Enviar al entorno productivo o de operación

Pruebas de Seguridad





**No Interrumpir cambios del
proceso de Desarrollo por
un bug o cambio de
emergencia**

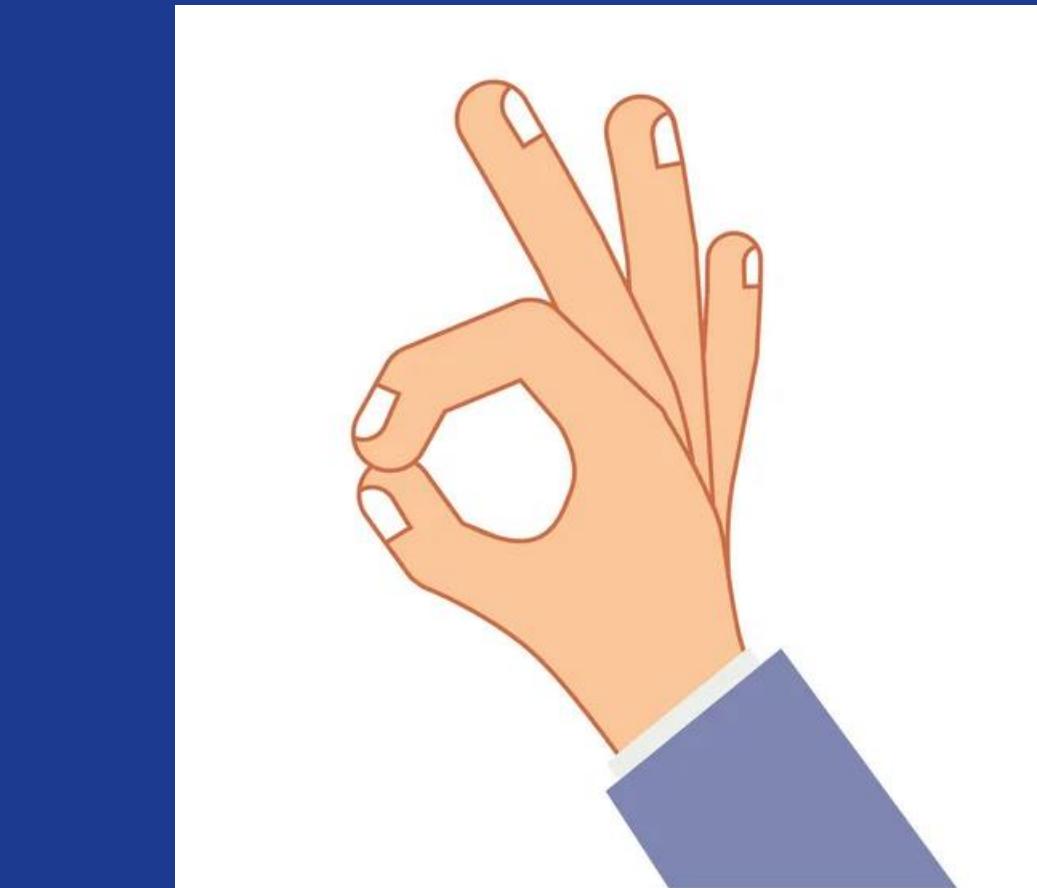
Reducción de riesgo

Verificar no dejar corrupto el proyecto

Establecer estándares mínimos de calidad

Establecer estándares mínimos de seguridad

No depender de acciones manuales



Partes del proceso



Code Management

Integración de código

Buenas prácticas:

- Tomar siempre la última versión antes de hacer un cambio.
- Hacer un merge local antes de subir cambios.
- Crear un Pull Request por Cambio
- Integrar cada commit a una tarea asignada o ticket
- Designar a alguien para la revisión de código

Opciones de manejo de versiones

Solo código y dependencias

Solo código

Código y Ejecutables



Agente de Compilación

Es un recurso de infraestructura que contiene un software especializado que puede realizar la compilación del código en un lenguaje en particular.

El agente se encarga de descargar el código de la rama solicitada y compilarla o interpretarla con la versión que se haya configurado.

El agente de compilación puede también encargarse de descargar dependencias, tanto de repositorios públicos como privados si así se configura.

El agente de compilación debe poder compilar múltiples proyectos de una misma solución.

Plugins y otros agentes

Para realizar otras tareas necesitamos otros tipos de agentes.

No necesariamente son agentes sino plugins sobre la herramienta que se esté utilizando.

Muchas tareas de estos plugins o agentes adicionales pueden realizarse en una máquina o servidor virtual dedicado a esto.

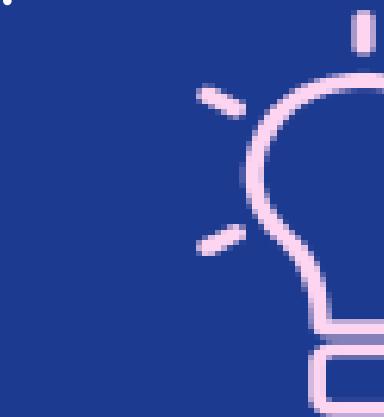
sonarQube



SMARTBEAR
ReadyAPI

Pero ... ¿A qué llamamos agente?

Es una máquina, con sistema operativo o una versión optimizada del mismo, que puede ser tanto física como virtual y contiene todo el software necesario para compilar una solución.



JVM, .NetFramework, GIT, Nuget, Maven, Programa para ejecutar pruebas, scanner de seguridad, etc.

Un agente en el contexto de desarrollo de software se refiere a un componente o conjunto de herramientas que automatizan ciertos procesos. Estos pueden incluir máquinas virtuales (VM) para la simulación de entornos de software, .NET Framework para la construcción de aplicaciones en Windows, GIT para el control de versiones, NuGet y Maven para la gestión de paquetes y dependencias de proyectos, programas para ejecutar pruebas automatizadas, y scanners de seguridad para detectar vulnerabilidades. Cada uno de estos componentes actúa como un agente que facilita y mejora el flujo de trabajo de desarrollo y despliegue de software.