



# Proyecto Certificador

## Desarrollo Software 3

Ficha de seguimiento N°3: Elaboración de la planificación del proyecto

### Indicador de logro N°2

Planifica la secuencia y duración de las actividades del proyecto en función al enfoque de Gestión de Proyectos, considerando los requisitos y artefactos a utilizar.

**Contenido**

1. Objetivo: .....3

2. Pasos:.....3

3. Instrumentos / Herramientas:.....3

4. Desarrollo.....4

## Ficha de seguimiento 3: Elaboración de la planificación del proyecto

*En esta ficha encontrarás los pasos para construir/trabajar esta parte de tu proyecto. Tienes un formato o plantilla para completar los datos y resultados de la actividad de esta ficha. Tu profesor/a revisará lo avanzado, y te brindará una retroalimentación.*

---

### Objetivo:

Se espera que el estudiante elabore de la planificación del proyecto, considerando dos pasos secuenciales, la Justificación de la elección del enfoque de Desarrollo de Software y ciclo de vida y luego la Planificación del Proyecto propiamente dicha.

### 1. Pasos:

Paso 1: Justificación de la elección del enfoque de Desarrollo de Software y ciclo de vida

Paso 2: Planificación del Proyecto

### 2. Instrumentos / Herramientas:

Manejo básico de las siguientes herramientas:

Procesador de texto MS-WORD o Google Docs

Presentaciones MS-POWER POINT o Google Docs

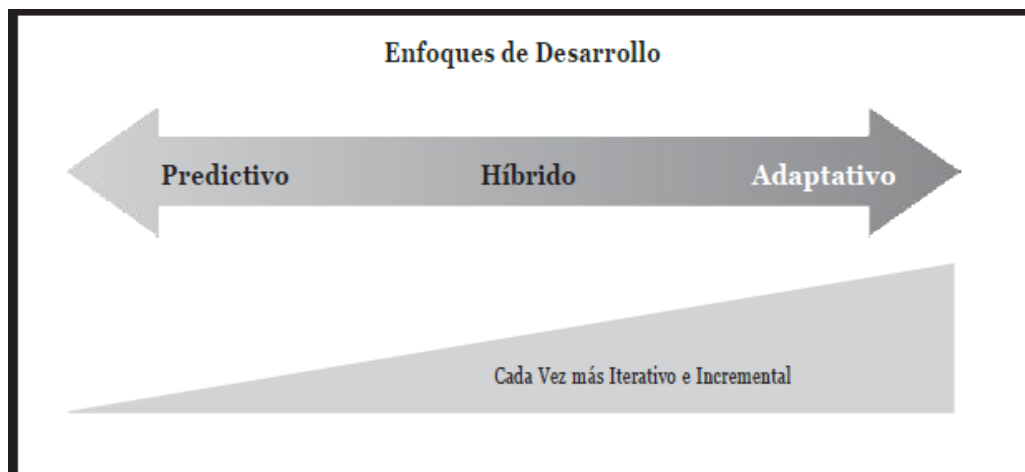
Hoja de Cálculo MS-EXCEL o Google Docs

### 3. Desarrollo

#### Paso 1: Justificación de la elección del enfoque de Desarrollo de Software y ciclo de vida

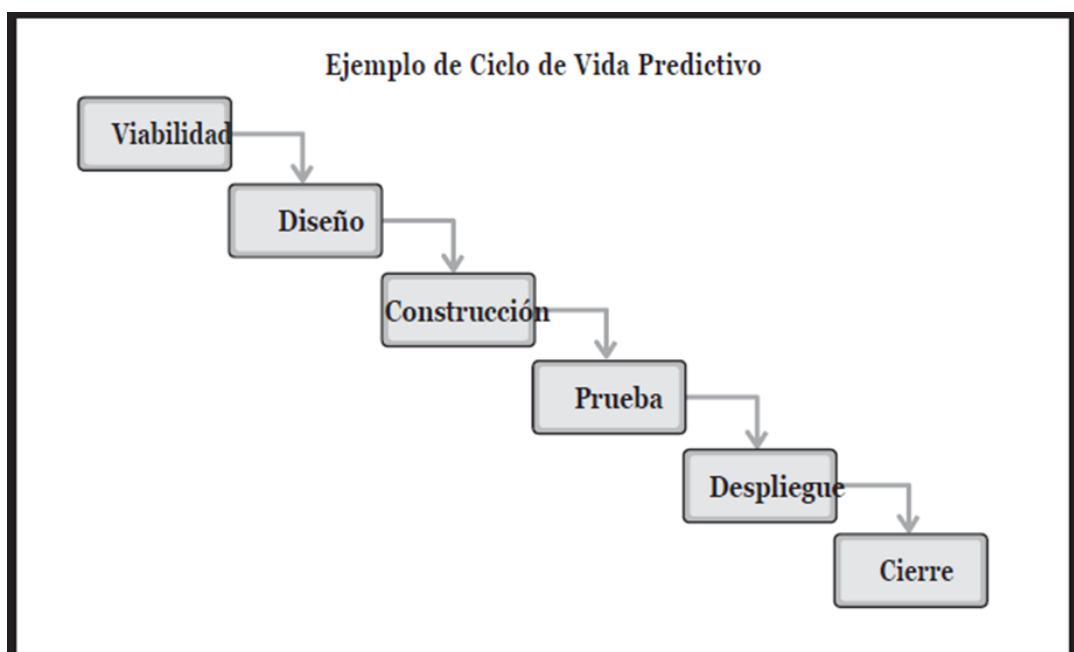
##### ***Enfoque de Gestión de Proyecto y Ciclo de vida a aplicar***

Se refiere a los diversos enfoques y las etapas que se usarán para planificar, ejecutar y controlar un proyecto de software.

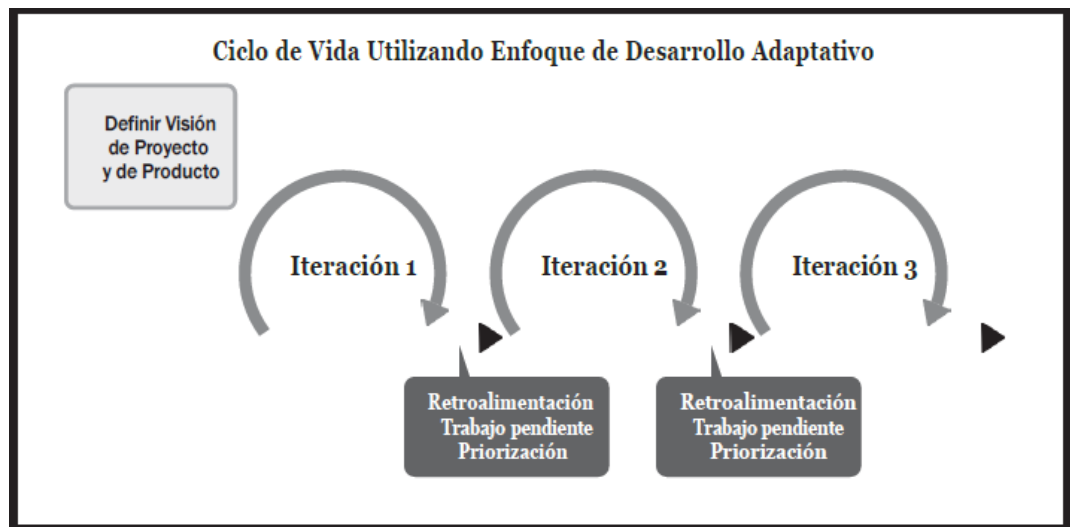


Enfoque de Gestión de Proyecto: Es la práctica que se utiliza para gestionar el proyecto. Algunos ejemplos son:

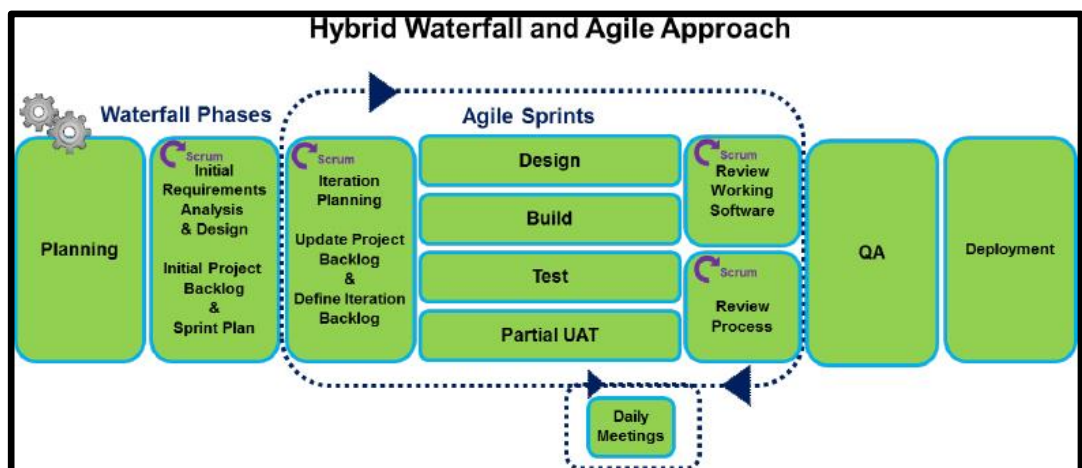
- ✓ Predictivo (Clásico o Cascada): Se caracteriza por un enfoque secuencial, donde cada etapa se completa antes de pasar a la siguiente.



- ✓ Adaptativo (Ágil): Se basa en la entrega incremental del producto, con ciclos cortos de desarrollo y pruebas.



- ✓ Híbrido: Combina elementos de los enfoques predictivo y adaptativo para gestionar proyectos.



### Ciclo de vida

Son las etapas que atraviesa un proyecto desde su inicio hasta su finalización. Un ciclo de vida típico se compone de las siguientes fases:

- ✓ Inicio: Se define el alcance del proyecto y se crean los documentos de planificación.
- ✓ Planificación: Se detalla el trabajo a realizar, se asignan recursos y se establece el cronograma.
- ✓ Ejecución: Se realizan las actividades necesarias para completar el proyecto.
- ✓ Seguimiento y control: Se monitorea el progreso del proyecto y se realizan ajustes si es necesario.
- ✓ Cierre: Se finalizan las actividades del proyecto y se documentan los resultados.

### ***Justificación del enfoque de gestión de Proyecto a aplicar***

La elección del enfoque de gestión de proyecto y del ciclo de vida dependerá de la naturaleza del proyecto, la complejidad del software, la experiencia del equipo y las preferencias del cliente.

Ejemplo: *"En este proyecto se utilizará un enfoque de gestión de proyecto Adaptativo (Ágil), con un ciclo de vida iterativo e incremental. Se implementarán las técnicas Scrum para la planificación del trabajo, el desarrollo del software y la colaboración entre los miembros del equipo". Y se justifica su uso por los siguientes beneficios:*

- 1. Flexibilidad y adaptabilidad: Scrum permite adaptar el proyecto a cambios en los requisitos o en el entorno del mercado. Los sprints cortos permiten una revisión y ajuste constante del proyecto, lo que reduce el riesgo de errores y aumenta la satisfacción del cliente.*
- 2. Mayor colaboración y comunicación: Scrum fomenta la comunicación y colaboración entre los miembros del equipo. Las reuniones diarias, las revisiones de sprint y las retrospectivas ayudan a mantener a todos informados y alineados con los objetivos del proyecto.*
- 3. Mayor transparencia y visibilidad: Scrum proporciona una vista clara del progreso del proyecto a través del backlog del producto, el sprint backlog y el burndown chart. Esto permite a los interesados (stakeholders) tener una mejor comprensión del estado del proyecto y tomar decisiones informadas.*
- 4. Mejora continua: Scrum incorpora la mejora continua como parte del proceso. Las retrospectivas al final de cada sprint permiten al equipo identificar áreas de mejora y tomar medidas para optimizar el proceso de desarrollo.*
- 5. Mayor motivación y compromiso del equipo: Scrum permite que el equipo autoorganice su trabajo y tome decisiones. Esto aumenta la responsabilidad y el compromiso del equipo con el éxito del proyecto.*

### ***Arquitectura de Software a Utilizar***

La arquitectura de software es la estructura y organización interna de un sistema de software, que define cómo se interconectan y trabajan juntos sus componentes para cumplir con los requisitos del sistema.

Ejemplos de algunos patrones de arquitectura actuales:

- ✓ Patrón de capas (Layer pattern): Divide el software en capas horizontales, cada una con una responsabilidad específica. Las capas comunes incluyen: Capa de presentación: Interfaz de usuario, Capa de aplicación: Lógica de negocio, Capa de acceso a datos: Acceso a la base de datos.
- ✓ Patrón cliente-servidor (Client-server pattern): Divide el software en dos partes: un cliente que solicita servicios y un servidor que los proporciona.

- ✓ Patrón MVC (Model-View-Controller): Separa la lógica de negocio (modelo), la presentación (vista) y la interacción del usuario (controlador).
- ✓ Microservicios (Microservices): Descompone el software en pequeños servicios independientes que se comunican entre sí a través de APIs.

### ***Modelos y artefactos a aplicar***

Debe describir que Modelos, Métodos y artefactos se van a emplear

#### Modelos:

- ✓ Modelo de cascada: Un enfoque lineal y secuencial para el desarrollo de software. Fases: Requerimientos, análisis, diseño, implementación, pruebas y mantenimiento. Adecuado para proyectos pequeños y bien definidos.
- ✓ Modelo en espiral: Un enfoque iterativo y incremental para el desarrollo de software. Combina el modelo de cascada con el desarrollo rápido de aplicaciones (RAD). Permite una mayor flexibilidad y adaptación a cambios en los requisitos.
- ✓ Modelo de prototipos: Se crea un prototipo funcional del software para probar y validar los requisitos. Permite una mejor comprensión del software por parte del cliente y los desarrolladores. Reduce el riesgo de errores en etapas posteriores del desarrollo.
- ✓ Modelo de desarrollo ágil: Se basa en la entrega incremental del software en ciclos cortos. Enfatiza la colaboración entre el cliente y el equipo de desarrollo. Se adapta mejor a proyectos con requisitos cambiantes o inciertos.

#### Artefactos:

- ✓ Especificaciones de requisitos: Documentos que describen los requisitos del software. Deben ser completos, precisos y comprensibles.
- ✓ Casos de uso: Descripciones de cómo los usuarios interactuarán con el software. Ayudan a comprender las necesidades del cliente y a validar el software.
- ✓ Diagramas de flujo: Representaciones visuales del flujo de la lógica del software. Ayudan a comprender el funcionamiento del software y a identificar posibles errores.
- ✓ Código fuente: El código escrito en un lenguaje de programación que define el comportamiento del software. Debe ser legible, mantenible y eficiente.
- ✓ Pruebas: Conjunto de actividades para verificar y validar el software. Ayudan a asegurar la calidad del software y a prevenir errores.

La elección de los modelos y artefactos adecuados dependerá de las necesidades específicas del proyecto, la metodología de desarrollo utilizada y las preferencias del equipo.

## Paso 2: Planificación del Proyecto

Corresponde al contenido que describe cómo se ejecutará, controlará y cerrará el proyecto. Este documento define el alcance, Objetivos del Proyecto (General y Específicos), el cronograma, Beneficios del Proyecto, interesados, supuestos, Factores Críticos de Éxito (FCE), Riesgos y Matriz de Comunicaciones

### ***Enunciado de Alcance del Proyecto (EAP)***

El Enunciado del Alcance del Proyecto es la descripción del alcance del proyecto, los principales entregables y las exclusiones. El Enunciado del Alcance del Proyecto (EAP) es un documento que describe de manera clara y concisa el alcance del proyecto. Es una herramienta fundamental para la gestión del proyecto, ya que define los límites de este y ayuda a asegurar que el equipo se enfoque en entregar lo que realmente se necesita.

En otras palabras, el EAP debe incluir:

- ✓ Una descripción del producto o servicio final del proyecto.
- ✓ Los principales entregables del proyecto.
- ✓ Las exclusiones del proyecto, es decir, lo que no se incluye en el mismo.

Ejemplo: Enunciado del Alcance del Proyecto (EAP) para un Proyecto de Desarrollo de Software Web

• Descripción del Proyecto: Desarrollar un software web para la gestión de inventario de una tienda de ropa. El software permitirá a los usuarios:

- ✓ Registrar la entrada y salida de productos:
  - Registrar la fecha, cantidad, precio y proveedor de los productos que ingresan al inventario.
  - Registrar la fecha, cantidad y precio de los productos que salen del inventario.
- ✓ Visualizar el stock actual de productos:
  - Mostrar el stock actual de cada producto en tiempo real.
  - Filtrar y ordenar los productos por diferentes criterios (categoría, marca, precio, etc.).
- ✓ Generar informes de inventario:
  - Generar informes en formato PDF y Excel con información sobre el stock actual, las entradas y salidas de productos, y el valor del inventario.
  - Personalizar los informes para incluir diferentes campos de información.
- ✓ Configurar alertas de stock bajo:
  - Configurar alertas que se activen cuando el stock de un producto alcance un nivel mínimo.
  - Definir el nivel mínimo de stock para cada producto.

• Entregables:

- ✓ Sitio web:



- Interfaz de usuario intuitiva y fácil de usar, con un diseño atractivo y adaptable a diferentes dispositivos.
- Funcionalidades completas para la gestión de inventario, según lo descrito en la sección 1.
- Informes de inventario en formato PDF y Excel, con la posibilidad de personalizarlos.
- Seguridad y confiabilidad, incluyendo medidas para proteger la información del inventario.

• Documentación:

- Manual de usuario que explique cómo usar el software web, con instrucciones claras y concisas.
- Guía técnica para desarrolladores que explique la arquitectura del software web y cómo realizar modificaciones.

• Exclusiones:

- Diseño gráfico personalizado: El diseño del sitio web se basará en una plantilla predefinida.
- Integración con sistemas de terceros: El software web no se integrará con otros sistemas en esta fase.
- Soporte técnico post-lanzamiento: Se ofrecerá un período de soporte técnico gratuito de 30 días después del lanzamiento del software web.

• Criterios de Aceptación:

- El software web debe cumplir con todos los requisitos funcionales y no funcionales especificados en este EAP.
- El software web debe ser compatible con los navegadores web más comunes (Chrome, Firefox, Safari, Edge).
- El software web debe ser seguro y confiable, y debe proteger la información del inventario.
- El software web debe ser fácil de usar y tener una interfaz de usuario intuitiva.
- La documentación del software web debe ser completa, precisa y fácil de entender.

• Restricciones:

- El proyecto tiene un presupuesto de \$20,000 USD.
- El proyecto tiene una duración de 15 semanas.
- El equipo de desarrollo está compuesto por 3 desarrolladores y 1 diseñador gráfico.

• Aprobaciones:

- Este EAP debe ser aprobado por el gerente de proyecto, el cliente y las partes interesadas.

### **Objetivos del Proyecto (General y Específicos)**

Los objetivos del proyecto son las metas que se quieren alcanzar al finalizar el mismo. Se dividen en dos tipos:

- **Objetivo general:** Describe la meta principal del proyecto de manera amplia y global. Debe ser específico, medible, alcanzable, relevante y con un plazo determinado (SMART). Ejemplo: Objetivo general: Desarrollar una nueva aplicación móvil para aumentar las ventas en un 20% en el próximo año.
- **Objetivos específicos:** Describen las metas específicas que se deben alcanzar para lograr el objetivo general. Deben ser concretos, medibles y verificables.

Ejemplos de Objetivos del Proyecto con redacción genérica:

- ✓ Objetivo específico 1: Diseñar una interfaz de usuario intuitiva y fácil de usar para la aplicación móvil.
- ✓ Objetivo específico 2: Implementar un sistema de pagos seguro y confiable dentro de la aplicación.
- ✓ Objetivo específico 3: Desarrollar una estrategia de marketing para la aplicación móvil que alcance a un público objetivo de 1 millón de usuarios.

En resumen, los objetivos del proyecto son las metas que se quieren alcanzar al finalizar el mismo. Es importante que sean SMART y que se definan tanto objetivos generales como específicos.

Ejemplos de los Objetivos del Proyecto previos ahora con redacción SMART:

Los objetivos SMART son específicos, medibles, alcanzables, relevantes y temporales. Se presenta una versión ajustada de cada uno de los objetivos previos:

- ✓ **Objetivo Específico 1**
  - Antes: Diseñar una interfaz de usuario intuitiva y fácil de usar para la aplicación móvil.
  - Después: Diseñar y probar una interfaz de usuario para la aplicación móvil, garantizando que el 90% de los usuarios de prueba puedan navegar por la aplicación sin dificultades y completar tareas clave en menos de cinco minutos. Este objetivo deberá cumplirse antes del 31 de diciembre de 2024.
- ✓ **Objetivo Específico 2**
  - Antes: Implementar un sistema de pagos seguro y confiable dentro de la aplicación.
  - Después: Implementar un sistema de pagos que cumpla con los estándares PCI DSS para seguridad de transacciones. Realizar pruebas de seguridad y obtener una tasa de éxito en transacciones del 99.5% antes del segundo trimestre de 2025.
- ✓ **Objetivo Específico 3**
  - Antes: Desarrollar una estrategia de marketing para la aplicación móvil que alcance a un público objetivo de 1 millón de usuarios.
  - Después: Desarrollar y lanzar una campaña de marketing digital para la aplicación móvil, con el objetivo de alcanzar a 1 millón de usuarios activos, medido a través de descargas y registros en la aplicación, dentro de los 12 meses posteriores al lanzamiento oficial de la aplicación.

- Estos objetivos refinados proporcionan claridad sobre lo que se espera lograr, cómo se medirá el éxito, asegurando que sean realistas y pertinentes, además de establecer una línea de tiempo definida para su realización.

### **Beneficios del Proyecto**

Se debe describir los resultados positivos que se esperan obtener al finalizar el mismo. Es decir, listar las principales ventajas que se obtendrán al completar el proyecto, un mínimo de tres y un máximo de cinco.

Algunos ejemplos de beneficios del proyecto son:

- ✓ Reducción de tiempos: El proyecto puede ayudar a reducir los tiempos operativos de la empresa.
- ✓ Reducción de costos: El proyecto puede ayudar a reducir los costos operativos de la empresa.
- ✓ Incremento de la capacidad: El proyecto puede ayudar a aumentar la capacidad operativa de los procesos.
- ✓ Incremento de la calidad: El proyecto puede ayudar a mejorar la calidad de los productos o servicios de la organización y por ello incrementar la satisfacción del cliente.

Es importante que los beneficios del proyecto sean específicos, medibles, alcanzables, relevantes y con un plazo determinado.

### **Cronograma**

El Cronograma del Proyecto es una representación puede ser tabular (tabla) o una gráfica (imagen) del tiempo que se espera que tome completar cada actividad del proyecto, generalmente se incluyen las principales actividades e hitos planificados. Se tiene El cronograma Planificado y también el Cronograma Ejecutado; para fines de seguimiento se presentan en un mismo diagrama de manera comparativa:

Ejemplo de Cronograma Planificado versus Cronograma Ejecutado:

**Cronograma Planificado y Ejecución al 30%**

| Fase                                | Fecha de Inicio Planificada | Fecha de Fin Planificada | % de Avance Planificado | % de Avance Real | Observaciones                                |
|-------------------------------------|-----------------------------|--------------------------|-------------------------|------------------|--|
| Planificación                       | 01/01/2024                  | 15/01/2024               | 10%                     | 10%              | Completado a tiempo.                         |
| Diseño de la Arquitectura           | 16/01/2024                  | 31/01/2024               | 20%                     | 20%              | Completado a tiempo.                         |
| Desarrollo de Interfaces de Usuario | 01/02/2024                  | 28/02/2024               | 25%                     | 0%               | Retraso debido a requisitos cambiantes.      |
| Implementación del Backend          | 01/03/2024                  | 31/03/2024               | 15%                     | 0%               | No iniciado, pendiente por retrasos previos. |
| Integración y Pruebas               | 01/04/2024                  | 30/04/2024               | 20%                     | 0%               | No iniciado, depende de fases anteriores.    |
| Lanzamiento                         | 01/05/2024                  | 15/05/2024               | 5%                      | 0%               | No iniciado.                                 |
| Mantenimiento y Actualizaciones     | 16/05/2024                  | Continuo                 | 5%                      | 0%               | No iniciado.                                 |

Los Interesados del Proyecto son individuos, grupos u organizaciones que pueden afectar, verse afectados o percibirse como afectados por una decisión, actividad o resultado del proyecto. En otras palabras, son cualquier persona o grupo que tenga un interés en el proyecto. Se pueden gestionar en base a un Registro de Interesados.

[illegible]