



## Ingeniería de software Modalidad Apilable

Guía para el profesor  
Clave PTTI2213  
Nivel Profesional Asociado

## Contenido

Datos generales.....	3
Competencia global .....	3
Introducción.....	3
Información general.....	4
Calendario de entregas tetramensual.....	8
Temario.....	9
Preguntas más frecuentes.....	11
Guía para las sesiones.....	12
Anexos.....	26
Rúbrica de la actividad integradora 1.....	26
Rúbrica de la actividad integradora 2.....	27
Rúbrica de la fase 1 proyecto .....	28
Rúbrica de la fase 2 proyecto .....	29
Prácticas de bienestar.....	30

## Datos generales

Nombre: Ingeniería de software

Nivel: Profesional Asociado

Modalidad: Apilable

Clave: PTTI2213

## Competencia global

Analiza y evalúa las metodologías para diseñar, desarrollar e implementar innovadoras y eficientes soluciones de software, que se ajusten a las necesidades específicas de empresas y organizaciones.

## Introducción

En esta experiencia educativa aprenderás los conceptos de ingeniería de software, el ciclo de vida, desde su concepción en ideas hasta su liberación y mantenimiento, se revisarán los principales modelos que existen para desarrollar software y las metodologías ágiles que se encuentran actualmente en el mercado.

Aprenderás sobre la ingeniería de requerimientos, las técnicas para identificar interesados, conocer sus fuentes información y definir requerimientos. Desde la perspectiva agile, aprenderás a desarrollar historias de usuario y gestionar el back log del producto.

En la etapa de diseño, aprenderás las técnicas para realizar un diseño detallado del software acorde a los requerimientos planteados en la etapa de análisis, aprenderás los diferentes modelos de arquitectura que existen con el objetivo de que aprendas a proponer arquitecturas que den solución a las necesidades del usuario. Estudiarás arquitecturas altamente demandadas en el mercado como la arquitectura de microservicios y revisarás elementos arquitectónicos muy relevantes como es la seguridad, la autenticación, autorización y el cifrado.

En la etapa de pruebas, aprenderás cómo desarrollar estrategias para probar el software, ejecución de pruebas funcionales, pruebas de seguridad, revisiones de código y monitoreo de la calidad del software.

Para la etapa de liberación, se revisarán los temas de gestiones de versiones, despliegue y mantenimiento del software.

En la etapa de mantenimiento, aprenderás sobre el innovador marco de trabajo DevOps que permite una mejor comunicación entre los equipos de desarrollo y operación utilizando herramientas como integración continua, contenedores y monitoreo.

Finalmente aprenderás la gestión del proyecto, cómo administrar los recursos humanos, materiales y técnicos, aprenderás diversas técnicas de estimación, administrar los riesgos y problemas en un proyecto.

## Información general

### Metodología

Un certificado **Apilable** ha sido diseñado con la finalidad de ser impartido a través de una metodología de flexibilidad para el aprendedor, ya que desde su diseño está estructurado para poder impartir a través de una modalidad autodirigida o bien en acompañamiento de un docente con experiencia en el ámbito laboral.

La experiencia de los **certificados apilables** promueve la interacción virtual entre aprendedores localizados en diferentes campus de la Universidad Tecmilenio como una forma de enriquecer tu formación contrastando la realidad de tu ciudad o región con la de otros compañeros, cuando así se lo permita la disponibilidad de este, considerando que podrás tener a su disposición la experiencia docente que enriquecerá su conocimiento. Sin embargo, se encuentran diseñados para ofrecer una experiencia autodirigida para aquellos aprendedores que por sus necesidades les demande ajustar sus propios tiempos.

#### Definiciones:

- I. **Apilabilidad:** modelo **nuevo de impartición** que puede realizarse bajo conducción de un académico o de manera autodirigida (el diseño del certificado tiene la flexibilidad de poder impartirse en ambos casos).
- II. **Duración:** 1 mes equivalente a 4 semanas efectivas.
- III. **Bajo conducción de un académico:** el contenido es impartido por un docente en sesiones sincrónicas o grabadas, en las cuales se abordarán los principales conceptos asociados a las unidades de aprendizaje. El profesor ofrece seguimiento y apoyo a los estudiantes. Estas sesiones virtuales sincrónicas de 9 horas a través de una herramienta tecnológica de videoconferencia, distribuidas de 2 a 3 sesiones por semana (de 3 a 4.5 horas por sesión). La asistencia a estas sesiones de videoconferencia es muy importante, pero en caso de no poder asistir, el aprendedor tiene la posibilidad de revisar la sesión grabada.
- IV. **Autodirigido:** son cursos asincrónicos sin un profesor asignado, con el contenido disponible a través de la plataforma de cursos (Canvas u otra). Los estudiantes disponen de todos los materiales para avanzar en su proceso de aprendizaje y la retroalimentación y evaluación se realiza entre pares o de forma automatizada en los casos que la plataforma lo permita.

### Bibliografía y software

Para cada módulo se sugiere la siguiente **bibliografía de apoyo**:

#### Bibliografía

D'andrade, B. (2020). *Software Engineering: Artificial Intelligence, Compliance, and Security*. Estados Unidos: Nova.

Wright, C. (2022). *Agile Project Management, Assurance and Auditing: A Practical Guide for Auditors, Reviewers and Project Teams*. Estados Unidos: IT Governance.

#### Software:

Tecmilenio. (s.f.). *Mi portal tecmilenio*. Recuperado de <https://miportal.tecmilenio.mx/> (Microsoft Project).

- Maestro, favor de explicar el proceso para descargar e instalar el software:
  - a. Ingresa a mi portal tecmilenio:



Pantalla 1. Página de inicio de mi portal tecmilenio.

- b. Haz clic en la lupa ubicada en la parte superior derecha de la página de inicio y escribe el nombre del software “*Microsoft Project*”:



Pantalla 2. Búsqueda del manual de instalación de *Microsoft Project*.

- c. Al hacer *enter* la página mostrará un manual de instalación de *Microsoft Project*. Haz clic sobre el icono del software que se muestra en la pantalla 2, esta acción permitirá descargar las instrucciones por seguir para la instalación de *Microsoft Project*.

## Evaluación

La evaluación es una combinación de los siguientes elementos:

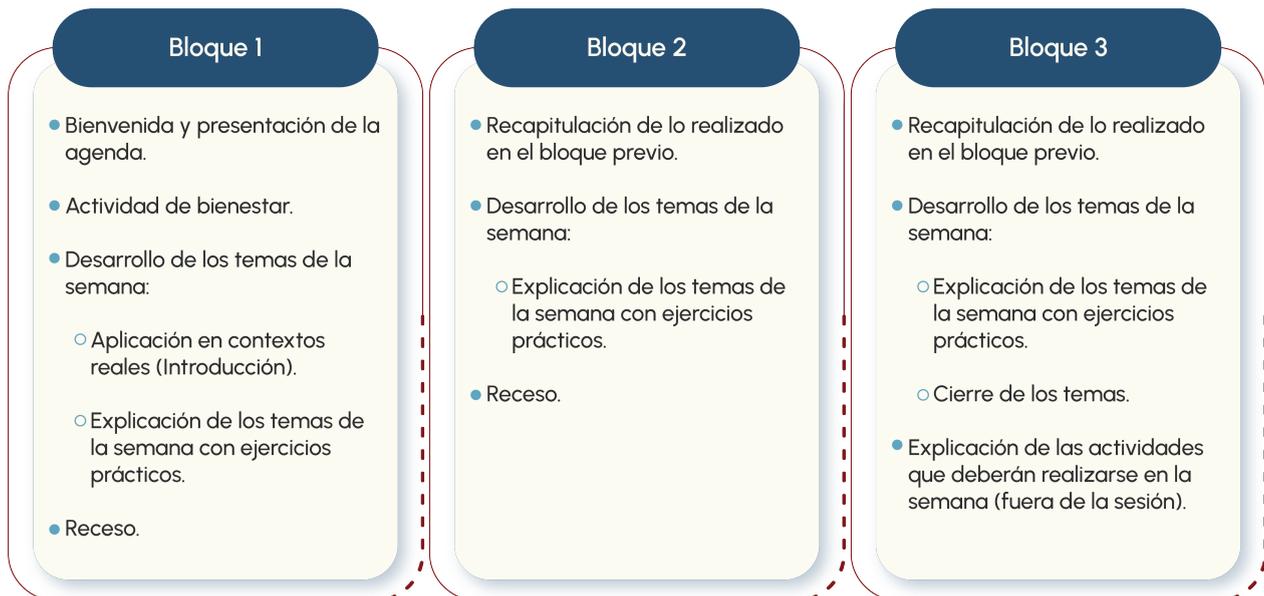
- Actividades que retoman el contenido conceptual de los temas revisados.
- Proyecto con el que el participante demostrará que adquirió las habilidades y los conocimientos requeridos para acreditar el certificado. Dicho proyecto se divide en dos fases.

A continuación, puedes revisar el detalle de la evaluación:

Evaluable	Ponderación
Actividad I	10%
Avance del proyecto	30%
Actividad II	10%
Entrega final del proyecto	40%
Examen final	10%
<b>Total</b>	<b>100%</b>

### Estructura de las sesiones

Las sesiones se dividen en tres bloques. Estas son las actividades que se recomienda realizar:



Antes de acudir a una sesión, es necesario que leas las explicaciones, ya que te proporcionarán los fundamentos teóricos de los temas. De igual manera, se requiere que revises las lecturas y los videos obligatorios.

Durante las sesiones sincrónicas, el docente da una breve explicación del tema, resuelve dudas y comparte las instrucciones de lo que se debe realizar fuera de dichas sesiones.

### Actividades y fases del proyecto

Las actividades, el avance y la entrega final del proyecto han sido diseñadas para realizarse de manera individual.

Como una forma de promover el dinamismo y la interacción de los participantes en distintos formatos, durante las sesiones, el docente alterna intervenciones individuales, plenarias y grupales que enriquecen tus puntos de vista y al mismo tiempo te dan la oportunidad de presentar tus ideas y posturas en torno a los temas de clase.

El resultado de todas las actividades, el avance y entrega final del proyecto realizadas deberá entregarse a través de la plataforma tecnológica para su revisión y evaluación por parte del docente.

Es muy importante que revises el esquema de evaluación y los criterios que utilizará el docente para otorgarte una calificación. Lo anterior con la intención de que desde el inicio de la semana tengas claro el nivel de complejidad y esfuerzo que requieres para realizar las entregas semanales y garantizar tu éxito dentro del certificado.

En caso de tener dudas sobre algunas actividades integradoras fases del proyecto o del contenido del certificado, puedes contactar a tu docente a través de los medios que te indique.

### Calendario de entregas tetramensual

Semana	Tema	Actividad integradora	Proyecto	Examen final
1	Tema 1. Fundamentos de la ingeniería de software	1		
	Tema 2. Ciclo de vida del software			
	Tema 3. Proceso del software			
	Tema 4. Ingeniería de software ágil			
	Tema 5. Proyectos de software vs. productos de software			
2	Tema 6. Ingeniería de requerimientos		Fase 1	
	Tema 7. Definición de requerimientos, características, escenarios e historias de usuario			
	Tema 8. Arquitectura de software			
	Tema 9. Software basado en la nube			
	Tema 10. Arquitectura de Microservicios			
3	Tema 11. Seguridad y privacidad	2		
	Tema 12. Calidad de software			
	Tema 13. Pruebas			
	Tema 14. DevOps y gestión del código			
	Tema 15. Despliegue y mantenimiento de software			
4	Tema 16. Procesos de desarrollo de software		Fase 2	X
	Tema 17. Gestión de proyectos - Parte 1			
	Tema 18. Gestión de proyecto - Parte 2			
	Tema 19. Ingeniería de software auxiliada por inteligencia artificial			
	Tema 20. Tendencias y retos futuros en la ingeniería de software			

## Temario

1. Fundamentos de la ingeniería de software
  - 1.1 Conceptos generales
  - 1.2 Ética en la ingeniería de software
  - 1.3 Casos prácticos de uso del software
2. Ciclo de vida del software
  - 2.1 Ciclo de vida del software
  - 2.2 Fases del ciclo de vida del software
3. Proceso del software
  - 3.1 Modelo cascada
  - 3.2 Modelo espiral
  - 3.3 Modelo incremental
  - 3.4 Modelo proceso evolutivo
4. Ingeniería de software ágil
  - 4.1 Métodos ágiles
  - 4.2 Programación extrema
  - 4.3 Scrum
5. Proyectos de software vs. productos de software
  - 5.1 La visión del producto
  - 5.2 Gestión de productos de software
  - 5.3 Prototipado de productos
6. Ingeniería de requerimientos
  - 6.1 Proceso de captura de requerimientos
  - 6.2 Proceso de análisis de requerimientos
  - 6.3 Habilidades del ingeniero de requerimientos
7. Definición de requerimientos, características, escenarios e historias de usuario
  - 7.1 Personas y escenarios
  - 7.2 Historias de usuario
  - 7.3 Identificación de características
8. Arquitectura de software
  - 8.1 Diseño y modelado de software
  - 8.2 Diseño arquitectónico
  - 8.3 Descomposición del sistema
  - 8.4 Arquitectura de distribución
  - 8.5 Modelos de arquitectura en software
9. Software basado en la nube
  - 9.1 Virtualización y contenedores
  - 9.2 Software como servicio
  - 9.3 Arquitectura de software en la nube
10. Arquitectura de Microservicios
  - 10.1 Microservicios y su arquitectura
  - 10.2 Servicios *restfull*
  - 10.3 Implementación del servicio
11. Seguridad y privacidad
  - 11.1 Ataques y defensas

- 11.2 Autenticación y autorización
- 11.3 Cifrado y privacidad
- 12. Calidad de software
  - 12.1 Definición de la calidad
  - 12.2 Modelos de calidad de software
  - 12.3 Métricas de calidad
- 13. Pruebas
  - 13.1 Pruebas funcionales
  - 13.2 Desarrollo basado en pruebas
  - 13.3 Pruebas de seguridad
  - 13.4 Revisiones de código
- 14. DevOps y gestión del código
  - 14.1 Integración y entrega continua (CI/CD)
  - 14.2 Contenedores y orquestación
  - 14.3 Monitoreo y *logging*
  - 14.4 Medición de DevOps
- 15. Despliegue y mantenimiento de software
  - 15.1 Despliegue del software características
  - 15.2 Mantenimiento de software
  - 15.3 Gestión de versiones
- 16. Procesos de desarrollo de software
  - 16.1 Modelos de desarrollo
  - 16.2 Gestión del cambio en la ingeniería de software
  - 16.3 Integración de modelos de madurez de capacidades (CMMI)
- 17. Gestión de proyectos - Parte I
  - 17.1 Definición del proyecto
  - 17.2 Planificación del proyecto
  - 17.3 Administración de personas y recursos
  - 17.4 Técnicas de estimación
  - 17.5 Cronograma del proyecto
- 18. Gestión de proyecto - Parte 2
  - 18.1 Técnicas de estimación
  - 18.2 Ejecución y control
  - 18.3 Administración de riesgos y calidad
  - 18.4 Comunicación y liderazgo
  - 18.5 Cierre y evaluación del proyecto
- 19. Ingeniería de software auxiliada por inteligencia artificial
  - 19.1 Aplicaciones IA para ingeniería de software
  - 19.2 Desarrollo de software con técnicas de IA
  - 19.3 Desafíos y oportunidades en la integración de IA
- 20. Tendencias y retos futuros en la ingeniería de software
  - 20.1 Ingeniería de software en la cuarta revolución industrial
  - 20.2 Ingeniería de software en la era de la computación cuántica

## Preguntas más frecuentes

### ¿En dónde o a quién le reporto un error detectado en el contenido?

Lo puedes reportar a través del botón “Mejora tu curso”, también puedes compartir sugerencias para el contenido y actividades del certificado.

### ¿Quién me informa de la cantidad de sesiones y el tiempo de cada sesión en las semanas?

El coordinador docente te debe proporcionar esta información.

### ¿En qué semanas se aplican los exámenes parciales y el examen final?

Consulta con tu coordinador docente los calendarios de acuerdo con la modalidad de impartición.

### ¿Tengo que capturar las calificaciones en Banner y en la plataforma educativa?

Sí, es importante que captures las calificaciones en la plataforma para que los participantes estén informados de su avance y reciban retroalimentación de parte tuya de todo lo que realizan en esta experiencia educativa. En Banner es el registro oficial de las calificaciones de los participantes.

## Guía para las sesiones

### Semana 1

#### Bloque 1

Actividad	Descripción	Duración
<b>Bienvenida y presentación de la agenda.</b>	El profesor se presenta ante el grupo y da una breve introducción de los temas que se abordarán.	5 minutos.
<b>Práctica de bienestar.</b>	El profesor impartidor seleccionará alguna práctica del banco anexo al final de este documento para compartirla en un foro de discusión y explicarla en la sesión. Se recomienda utilizar una diferente por semana.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ <b>Aplicación en contextos reales (introducción).</b></li> <li>○ <b>Explicación de los temas de la semana con ejercicios prácticos.</b></li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	40 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

#### Bloque 2

Actividad	Descripción	Duración
<b>Resumen de lo realizado en el bloque anterior.</b>	El profesor resumirá de manera dinámica lo realizado en el bloque anterior.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ <b>Explicación de los temas de la semana con ejercicios prácticos.</b></li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	45 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

### Bloque 3

Actividad	Descripción	Duración
<b>Recapitulación de lo realizado en el bloque previo.</b>	El profesor recapitulará de manera dinámica lo realizado en el bloque previo.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ <b>Explicación de los temas de la semana con ejercicios prácticos.</b></li> <li>○ <b>Cierre de los temas.</b></li> </ul>	<ul style="list-style-type: none"> <li>● El profesor explicará a los participantes los contenidos con ejercicios prácticos y realizará un cierre de los temas correspondientes.</li> </ul>	35 minutos.
<b>Explicación de la actividad integradora I.</b> <b>Explicación del proyecto, con enfoque en la fase I.</b>	<p>Se explicará a los participantes en qué consiste la actividad integradora I, la cual se entrega en la semana 1.</p> <p>Se explicará a los participantes en qué consiste el reto de manera general, enfocándose en la fase I, la cual deberán entregar en la semana 2.</p>	10 minutos.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 1 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar el ciclo de vida del *software*: análisis, diseño, construcción, pruebas, implementación y mantenimiento.
- Explicar que la ingeniería de software proporciona métodos y técnicas para el desarrollo de software confiable y de calidad, incluida la estimación del tiempo y los costos de desarrollo.
- Explicar que se necesitan métodos y técnicas para identificar las necesidades de los usuarios, seleccionar tecnologías apropiadas, formar un equipo capacitado y estimar el tiempo y el costo de un proyecto.
- Explicar que los principios éticos en la ingeniería de software incluyen considerar el interés público, ofrecer soluciones de alta calidad, evitar conflictos de intereses y promover un comportamiento ético.
- Explicar que existen siete categorías de software: software de sistemas, software de aplicaciones, software de ingeniería y ciencia, software integrado, software de línea de productos, aplicaciones web y software de inteligencia artificial.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 2 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que el ciclo de vida del software es una estructura de procesos y actividades relacionadas con el desarrollo y mantenimiento de un producto de software.
- Explicar la importancia de conocer el ciclo de vida para evitar errores en áreas como la experiencia de usuario, seguridad, eficiencia, estabilidad, costos y calidad.

- Explicar por qué es necesaria una buena comunicación entre el cliente y el equipo de desarrollo para definir los requerimientos de forma clara y consistente.
- Explicar las fases del ciclo de vida del software que son: planificación, análisis, diseño, codificación, pruebas, despliegue y mantenimiento.
- Explicar la importancia de establecer los objetivos de cada fase del ciclo de vida del software, así como la visión del producto, comprender las necesidades del cliente, tomar decisiones de diseño, desarrollar el código, realizar pruebas, implementar y monitorear.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 3 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que el modelo de cascada, también conocido como ciclo de vida clásico, propone un enfoque sistemático y secuencial para el desarrollo de software.
- Explicar la importancia del modelo de cascada y por qué es adecuado para proyectos donde los requisitos son fijos o se requieren cambios mínimos.
- Explicar las ventajas del modelo de cascada entre las que se encuentran la claridad y definición de los pasos, así como la promoción de buenos hábitos de codificación.
- Explicar las desventajas del modelo de cascada, tales como la dificultad para realizar cambios durante el proceso del proyecto y la poca participación del usuario o cliente final.
- Explicar la fase de pruebas que se encuentra al final del proyecto en el modelo de cascada, lo cual aumenta los riesgos de errores cometidos anteriormente.
- Explicar por qué el modelo de cascada es el paradigma más antiguo de la ingeniería de software y no todos los proyectos se ajustan a un flujo secuencial.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 4 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que los métodos ágiles combinan una filosofía y un conjunto de lineamientos de desarrollo, enfocándose en la satisfacción del cliente y la entrega rápida de software incremental.
- Explicar el "Manifiesto por el desarrollo agile de software" que establece los principios fundamentales de la agilidad en el desarrollo de software.
- Explicar qué es la agilidad y qué implica la capacidad de responder adecuadamente a los cambios y reconocer la importancia de la colaboración y comunicación efectiva en equipos de desarrollo.
- Explicar qué buscan los procesos ágiles con respecto a los requerimientos que pueden cambiar a lo largo del proyecto, tomando en cuenta un enfoque empírico y adaptable en el diseño, construcción y pruebas.
- Explicar los 12 principios de agilidad que se buscan alcanzar en los proyectos de desarrollo.
- Explicar qué es la programación extrema (XP) y en qué está basada.
- Explicar por qué la comunicación entre clientes y desarrolladores es esencial en la programación extrema para establecer características y funciones requeridas para el software.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 5 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar qué es la visión del producto, al ser una declaración que explica las necesidades del negocio y debe ser flexible para adaptarse a los cambios.
- Explicar cuáles son los principales errores del Product Owner.
- Explicar que los objetivos de tener una visión del producto son generar inspiración, mejorar la transparencia, habilitar la inspección y adaptación, y relacionar el contexto y el mercado.
- Explicar la importancia de definir historias de usuarios, que capturan los requisitos del cliente.

**Notas para la actividad integradora 1.**

1. Si el aprendiz lo requiere, debe profundizarse en los diferentes temas para generar los conocimientos necesarios.
2. Mostrar un ejemplo de código para instruir al aprendiz sobre qué se espera de su actividad.
3. Apoyar al aprendiz en la generación del *road map* y el prototipo, así como en aquello que se espera de su entregable de la actividad integradora 1.
4. En conjunto con los aprendices, leer la descripción de los entregables esperados para aclarar cualquier duda que pudiera surgir al respecto.
5. La actividad integradora 1 abarca del tema 1 al 5.

Se entrega en la semana 1.

**Notas para el profesor impartidor. Estas corresponden a la explicación del proyecto, con enfoque en las dos fases en que se divide, para que tengan tiempo de revisarlo; sin embargo, se ahonda en la entrega de la fase 1.**

Al profesor impartidor, se le recomienda lo siguiente:

1. Para realizar esta parte del proyecto, se recomienda que el aprendiz posea conocimientos referentes a los diez primeros temas.
2. El objetivo de esta fase del proyecto es que el aprendiz, mediante la programación en Java, resuelva un problema de la vida real; para ello, deberá utilizar la lógica, sus conocimientos en lenguaje y estructuras de datos lineales, así como los algoritmos.
3. Se recomienda que el profesor elabore una serie de ideas sobre cómo lograr la solución del problema mediante la lógica de programación.
4. Si es necesario, el profesor puede hacer una demostración de un código complejo y centrarse en superar el problema específico de una empresa. De esta manera, el aprendiz observará una solución parecida al entregable del proyecto.
5. Realiza la evaluación de la fase 1 del proyecto, con base en los criterios de la rúbrica, y proporciona la retroalimentación del entregable. Se recomienda que el profesor destaque las fortalezas analíticas del aprendiz.
6. Lleva a cabo una retroalimentación positiva basada en la técnica del sándwich (lo que está bien, lo que debe ser mejorado y cómo se puede corregir).

Se entrega en la semana 2.

## Semana 2 Bloque 1

Actividad	Descripción	Duración
<b>Bienvenida y presentación de la agenda.</b>	El profesor da una breve bienvenida y presenta la agenda de la sesión.	5 minutos.
<b>Práctica de bienestar.</b>	El profesor impartidor seleccionará alguna práctica del banco anexo al final de este documento para compartirla en un foro de discusión y explicarla en la sesión. Se recomienda utilizar una diferente por semana.	5 minutos.

<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Aplicación en contextos reales (introducción).</li> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	30 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

### Bloque 2

Actividad	Descripción	Duración
<b>Recapitulación de lo realizado en el bloque previo.</b>	El profesor recapitulará de manera dinámica lo realizado en el bloque previo.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	45 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

### Bloque 3

Actividad	Descripción	Duración
<b>Recapitulación de lo realizado en el bloque previo.</b>	El profesor recapitulará de manera dinámica lo realizado en el bloque previo.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> <li>○ Cierre de los temas.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos y realizará un cierre de los temas correspondientes.	35 minutos.
<b>Explicación de la actividad integradora 2.</b>  <b>Recordatorio de entrega de la fase 1 del proyecto.</b>	Se explicará a los participantes en qué consiste la actividad integradora 2 que se entrega en la semana 3.  Se recordará a los participantes en qué consiste la fase 1 del proyecto, la cual deberán entregar en la semana 2.	10 minutos.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 6 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que el proceso de captura de requerimientos es crucial en el desarrollo de proyectos, ya que permite comprender las necesidades, expectativas y restricciones de los involucrados.
- Explicar que el proceso de captura de requerimientos consta de 6 etapas principales y describirlas.
- Explicar que el proceso de análisis de requerimientos tiene etapas principales y describir cada una de ellas.
- Explicar por qué un ingeniero de requerimientos debe tener habilidades tales como: pensamiento analítico, empatía, habilidades de comunicación, resolución de conflictos, moderación, confianza en sí mismo y persuasión.
- Explicar por qué estas etapas y habilidades son fundamentales para garantizar que los requerimientos sean claros, completos y factibles de implementar antes de pasar a la fase de diseño y construcción del proyecto.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 7 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar la importancia de desarrollar el perfil de una persona como un personaje ficticio que representa los deseos, necesidades y rasgos de un subconjunto del público objetivo del sistema.
- Explicar la importancia de un mapa de empatía, que es una herramienta visual que ayuda a comprender mejor las necesidades, deseos, motivaciones y frustraciones de los usuarios o clientes.
- Explicar qué aspectos incluye un mapa de empatía, y que contiene campos que incluyen información sobre el cliente objetivo, sus pensamientos y sentimientos, sus necesidades, frustraciones y objetivos.
- Explicar que una historia de usuario es una explicación general e informal de una función de software desde la perspectiva del usuario final.
- Explicar cómo se expresan las historias de usuario, las cuales pueden incluir el perfil del usuario, la actividad que desea realizar y el valor que busca obtener.
- Explicar los criterios de aceptación de las historias de usuario y cuáles que existen algunas herramientas como JIRA y Trello para gestionar el backlog del producto y las historias de usuario.
- Explicar qué es el modelo INVEST y por qué es una guía para escribir buenas historias de usuario, que sean independientes, negociables, valiosas, estimables, pequeñas y testeables.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 8 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar las ventajas y desventajas del diseño y modelado, enfatizando cómo facilita la planificación, la detección de errores tempranos y la claridad en la comunicación del equipo.
- Explicar el proceso de diseño arquitectónico y su importancia en la estructura general del sistema.
- Utilizar un ejemplo práctico, como el sistema de gestión de biblioteca, para explicar el proceso de descomposición en pasos claros.
- Introducir el concepto de arquitectura de distribución y su objetivo principal de escalabilidad y robustez.
- Presentar diferentes modelos de arquitectura de software, como MVC, Cliente-Servidor, Capas, Microservicios, y Arquitectura Hexagonal.

- Explicar las características y ventajas de cada modelo, utilizando ejemplos prácticos para ilustrar su aplicación en el desarrollo de software.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 9 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Comenzar con una explicación clara de qué es la virtualización y cómo ha revolucionado la utilización del hardware físico mediante la creación de máquinas virtuales (VMs).
- Utilizar la tabla de comparación proporcionada para explicar las diferencias clave entre virtualización y contenedores en términos de nivel de abstracción, aislamiento, rendimiento, tamaño de imagen, tiempo de inicio, manejabilidad y uso de recursos.
- Discutir las ventajas de los contenedores, como su eficiencia y portabilidad, y las desventajas como el menor aislamiento comparado con las VMs.
- Explicar las diferencias entre SaaS, PaaS (Platform as a Service) e IaaS (Infrastructure as a Service) utilizando la tabla de comparación proporcionada.
- Proporcionar ejemplos prácticos y casos de uso de cómo la arquitectura en la nube se implementa en la industria, destacando sus ventajas y desafíos.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 10 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar qué son los microservicios, destacando que es un estilo arquitectónico y un modo de programar software que estructura una aplicación como una colección de servicios pequeños y autónomos.
- Describir los principios clave, como la descentralización, independencia, heterogeneidad tecnológica, resiliencia, automatización y monitoreo.
- Explicar las ventajas de los microservicios, como modularidad, escalabilidad, versatilidad y facilidad de depuración y pruebas.
- Discutir las desventajas, incluyendo la complejidad en la comunicación y gestión, el mayor consumo de memoria y la mayor complejidad en la gestión de la configuración.
- Proporcionar ejemplos prácticos de servicios RESTful, como APIs de redes sociales, servicios en la nube y servicios de mapas y localización.
- Detallar el proceso de definir endpoints del API, desarrollar la lógica del negocio, usar un framework de desarrollo, implementar seguridad, realizar pruebas y desplegar el microservicio en un entorno de cloud como AWS o Azure.

**Notas para la actividad integradora 2.**

Al profesor impartidor, se le recomienda lo siguiente:

1. Si el aprendiz lo requiere, debe profundizarse en los diferentes métodos de ordenamiento para generar los conocimientos necesarios.
2. Mostrar un ejemplo de código para instruir al aprendiz sobre qué se espera de su actividad.
3. Apoyar al aprendiz en la generación de diagramas de flujo y pseudocódigo, así como en lo que se espera de su entregable de la actividad integradora 2.
4. En conjunto con los aprendices, leer la descripción de entregables y la rúbrica para aclarar cualquier duda que pudiera surgir al respecto.
5. La actividad integradora 2 abarca del tema 6 al 15.

Se entrega en la semana 3.

Recordatorio de la entrega del proyecto, fase 1.

### Semana 3

#### Bloque 1

Actividad	Descripción	Duración
<b>Bienvenida y presentación de la agenda.</b>	El profesor da una breve bienvenida y presenta la agenda de la sesión.	5 minutos.
<b>Práctica de bienestar.</b>	El profesor impartidor seleccionará alguna práctica del banco anexo al final de este documento para compartirla en un foro de discusión y explicarla en la sesión. Se recomienda utilizar una diferente por semana.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Aplicación en contextos reales (introducción).</li> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	30 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

#### Bloque 2

Actividad	Descripción	Duración
<b>Recapitulación de lo realizado en el bloque previo.</b>	El profesor recapitulará de manera dinámica lo realizado en el bloque previo.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	45 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

### Bloque 3

Actividad	Descripción	Duración
<b>Rescapitulación de lo realizado en el bloque previo.</b>	El profesor recapitulará de manera dinámica lo realizado en el bloque previo.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>o <b>Explicación de los temas de la semana con ejercicios prácticos.</b></li> <li>o <b>Cierre de los temas.</b></li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos y realizará un cierre de los temas correspondientes.	35 minutos.
<b>Explicación del proyecto, con enfoque en la fase 2.</b>  <b>Recordatorio de entrega de la actividad integradora 2.</b>	<p>Se explicará a los participantes en qué consiste el proyecto de manera general, enfocándose en la fase 2 la cual deberán entregar en la semana 4.</p> <p>Se recordará a los participantes en qué consiste la actividad integradora 2, la cual se entrega en la semana 3.</p>	10 minutos.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 11 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Comenzar con una breve historia de los ataques cibernéticos, mencionando la evolución desde los primeros virus informáticos en los años 80 hasta las sofisticadas amenazas actuales como el *phishing*, *ransomware* y APT (Amenazas Persistentes Avanzadas).
- Explicar los riesgos asociados con los ataques informáticos, incluyendo la pérdida de información confidencial, daños financieros, interrupción de operaciones y daño a la reputación de las empresas.
- Destacar la importancia de la formación y concienciación de los usuarios sobre las amenazas de seguridad y cómo evitarlas.
- Discutir mejores prácticas como la implementación de MFA (Autenticación de Múltiples Factores), el uso de protocolos estándar como OAuth y OpenID Connect, y el principio de menor privilegio.
- Proporcionar mejores prácticas para la implementación de cifrado robusto, gestión segura de claves, minimización de datos, transparencia y cumplimiento normativo.
- Fomentar la discusión y el intercambio de ideas entre los estudiantes para mejorar su comprensión y habilidades en la implementación de medidas de seguridad.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 12 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar la importancia de la calidad en la ingeniería de software, destacando su impacto en la satisfacción del usuario, la seguridad y la sostenibilidad del software a lo largo de su ciclo de vida.

- Explicar cómo cada modelo aborda la calidad desde diferentes perspectivas: calidad de proceso, calidad de producto y calidad de uso.
- Discutir cómo las métricas de calidad ayudan a mejorar la satisfacción del cliente, reducir costos, aumentar la eficiencia y fomentar la mejora continua.
- Proporcionar ejemplos de cómo las métricas pueden ser utilizadas para monitorear y mejorar la calidad del software durante su desarrollo y mantenimiento.
- Discutir la adopción de metodologías ágiles, DevOps y la integración continua como enfoques modernos para asegurar la calidad del software desde las primeras etapas del desarrollo.
- Fomentar la discusión y el intercambio de ideas entre los estudiantes para mejorar su comprensión y habilidades en la implementación de modelos y métricas de calidad del software.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 13 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar la evolución desde el modelo de cascada hacia un enfoque iterativo e incremental en el desarrollo de software, destacando la importancia de integrar las pruebas funcionales desde las primeras etapas del proyecto.
- Definir las pruebas funcionales como esenciales para validar que el software ejecuta correctamente las funciones especificadas en los requisitos.
- Explicar la importancia de las pruebas de seguridad para identificar vulnerabilidades, amenazas y riesgos en el software.
- Describir los tipos principales de pruebas de seguridad.
- Describir los tipos de revisiones de código (formales e informales) y sus objetivos principales.
- Fomentar la discusión y el intercambio de ideas entre los estudiantes para mejorar su comprensión y habilidades en la realización de pruebas de software.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 14 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar qué son la Integración Continua (CI) y la Entrega Continua (CD) y cómo estas prácticas mejoran y agilizan el ciclo de vida de desarrollo del software.
- Mencionar cómo la CI/CD facilita la automatización de pruebas y despliegues, permitiendo que el software esté siempre en un estado desplegable.
- Describir el proceso de Integración Continua, incluyendo la integración frecuente de código en el repositorio y la automatización de pruebas.
- Proporcionar ejemplos de herramientas populares para CI/CD, como Jenkins, GitLab CI/CD, Travis CI, y CircleCI.
- Discutir los beneficios de la orquestación de contenedores, como la automatización completa, la gestión eficiente de recursos, la escalabilidad y el balanceo de carga.
- Fomentar la discusión y el intercambio de ideas entre los estudiantes para mejorar su comprensión y habilidades en la implementación de prácticas de DevOps.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 15 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que el despliegue de software implica transferir aplicaciones desde el ambiente de desarrollo hasta producción.
- Explicar los diferentes tipos de estrategias de despliegue que son: el despliegue manual y el despliegue automatizado.
- Explicar qué es el despliegue continuo, mencionando que es una extensión de la entrega continua, donde los cambios se liberan automáticamente sin intervención manual.
- Explicar lo que hace el despliegue basado en contenedores al empaquetar las aplicaciones y sus dependencias en contenedores portátiles y ligeros.

- Explicar que el mantenimiento de software incluye actividades como la corrección de errores, el mantenimiento evolutivo y el mantenimiento preventivo.
- Explicar que el mantenimiento preventivo busca prevenir problemas futuros identificando y eliminando fuentes de error o debilidades en el código.
- Explicar que la gestión de versiones permite registrar y recuperar versiones específicas de archivos a lo largo del tiempo y facilita la colaboración entre equipos.

### Notas para la fase del proyecto, con enfoque en la fase 2.

Al profesor impartidor, se le recomienda lo siguiente:

1. Para realizar esta parte del proyecto, se recomienda que el aprendedor posea los conocimientos referentes a métodos de ordenamiento y búsqueda, los cuales se revisan en los temas finales del curso.
2. El objetivo de esta fase del proyecto es que el aprendedor, mediante la programación en Java, mejore sus implementaciones.
3. Se aconseja que el profesor explique detalladamente cómo puede mejorarse el código entregado en la fase 1, gracias a la implementación de métodos de búsqueda, ordenamiento y grafos.
4. De ser necesario, el profesor puede hacer una demostración de un código complejo y centrarse en superar el problema de una empresa. De esta manera, el aprendedor observará una solución parecida al entregable del proyecto, fase 2.
5. Realiza la evaluación de la fase 2 del proyecto, con base en los criterios de la rúbrica, y proporciona la retroalimentación del entregable. Se recomienda que el profesor subraye las fortalezas analíticas del aprendedor.
6. Realiza una retroalimentación positiva, basada en la técnica del sándwich (lo que está bien, lo que debe ser mejorado y cómo se puede corregir).

Se entrega en la semana 4.

Recordatorio de entrega de la actividad integradora 2.

## Semana 4 Bloque 1

Actividad	Descripción	Duración
<b>Bienvenida y presentación de la agenda.</b>	El profesor da una breve bienvenida y presenta la agenda de la sesión.	5 minutos.
<b>Práctica de bienestar.</b>	El profesor impartidor seleccionará alguna práctica del banco anexo al final de este documento para compartirla en un foro de discusión y explicarla en la sesión. Se recomienda utilizar una diferente por semana.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ <b>Aplicación en contextos reales (introducción).</b></li> <li>○ <b>Explicación de los temas de la semana con ejercicios prácticos.</b></li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	40 minutos.

<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.
----------------	--	-------------

### Bloque 2

Actividad	Descripción	Duración
<b>Resumen de lo realizado en el bloque anterior.</b>	El profesor resumirá de manera dinámica lo realizado en el bloque anterior.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos.	45 minutos.
<b>Receso.</b>	Se brindará un espacio de receso para que el participante lo utilice a su beneficio.	10 minutos.

### Bloque 3

Actividad	Descripción	Duración
<b>Resumen de lo realizado en el bloque anterior.</b>	El profesor resumirá de manera dinámica lo realizado en el bloque anterior.	5 minutos.
<b>Desarrollo de los temas de la semana:</b> <ul style="list-style-type: none"> <li>○ Explicación de los temas de la semana con ejercicios prácticos.</li> <li>○ Cierre de los temas.</li> </ul>	El profesor explicará a los participantes los contenidos con ejercicios prácticos y realizará un cierre de los temas correspondientes.	35 minutos.
<b>Recordatorio de entrega del proyecto, fase 2.</b>	Se recordará a los participantes en qué consiste la fase 2 del proyecto, la cual deberán entregar en la semana 4.	10 minutos.
<b>Recordatorio del examen final.</b>	Recordar el examen final en la semana 4.	

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 16 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar las dos categorías principales de modelos: metodologías ágiles y metodologías tradicionales.
- Explicar que las metodologías tradicionales siguen un enfoque secuencial y estructurado, como el modelo en cascada y el modelo en espiral.

- Explicar que las metodologías ágiles surgieron como respuesta a las limitaciones de los enfoques tradicionales y se centran en la flexibilidad, colaboración y entrega continua de valor.
- Explicar por qué la gestión del cambio en la ingeniería de software es crucial para implementar adaptaciones y mejoras de manera eficiente.
- Explicar que el proceso de gestión del cambio incluye la identificación del cambio, evaluación y aprobación, planificación, implementación, y revisión y evaluación.
- Explicar cuáles son las herramientas y técnicas utilizadas en la gestión del cambio, que incluyen los sistemas de gestión de cambios, las metodologías ágiles y la comunicación y formación.
- Explicar qué es el modelo de madurez de capacidades CMMI y cómo ayuda a las organizaciones a desarrollar procesos más eficientes y predecibles.
- Explicar los cinco niveles de madurez del CMMI: Inicial, Gestionado, Definido, Cuantitativamente gestionado y Optimizado.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 17 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explica que la planificación del proyecto ayuda en la creación de planes para alcanzar los objetivos, incluyendo el plan de gestión del proyecto, el alcance, el cronograma y los riesgos.
- Explica que la administración de personas y recursos implica gestionar eficazmente los recursos humanos, materiales, financieros y tecnológicos.
- Explica qué es la gestión de recursos humanos y los aspectos que incluye, como la selección del equipo, desarrollo de habilidades, motivación y resolución de conflictos.
- Explica qué es la gestión de recursos materiales y financieros, y los aspectos que requiere: una planificación cuidadosa y un seguimiento para utilizar eficientemente los recursos y mantenerse dentro del presupuesto.
- Explica las estrategias efectivas en la gestión de recursos que incluyen la planificación detallada, comunicación continua y monitorización y ajustes.
- Explica en qué consisten las técnicas de estimación y sus aspectos, como son: juicio de expertos, paramétrica, por analogía, ascendente y simulación.
- Explica qué es un cronograma del proyecto, enfatizando que es una herramienta esencial para gestionar el tiempo y la coordinación de recursos, a través del uso de diagramas de Gantt, el establecimiento de hitos y de un software de gestión de proyectos.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 18 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar las técnicas de estimación de costos de proyectos y cuáles son: estimación analógica, estimación paramétrica, estimación Bottom-Up y estimación de tres puntos.
- Explicar en qué consiste la fase de ejecución y control del proyecto y que depende de la capacidad del gestor del proyecto para motivar y liderar al equipo, manteniendo altos niveles de moral y compromiso.
- Explicar en qué consiste el seguimiento y control del progreso, así como los elementos que son importantes llevar a cabo en esta fase.
- Explicar cómo se lleva a cabo la fase de gestión del cambio.
- Explicar cuáles son las herramientas y estrategias para llevar a cabo un control efectivo dentro del proyecto.
- Explicar cómo se lleva a cabo la gestión de riesgos del proyecto y cómo se puede evaluar a través de la matriz de evaluación de probabilidad e impacto del riesgo.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 19 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que las aplicaciones de IA en la ingeniería de software incluyen la automatización de pruebas, el análisis de código asistido por IA y la optimización de la gestión de proyectos mediante IA.
- Explicar que la automatización de pruebas de software potenciada por IA permite reducir el tiempo necesario para las pruebas y aumentar la cobertura de código.
- Explicar que el análisis de código asistido por IA ayuda a identificar problemas de seguridad y mejorar la calidad del código.
- Explicar que la IA puede optimizar la gestión de proyectos al prevenir retrasos, estimar costos y optimizar la asignación de recursos.
- Explicar que la asistencia avanzada en la codificación con IA acelera el desarrollo de software y reduce la tasa de errores.
- Explicar que la IA puede mejorar la experiencia del usuario al analizar interacciones y sugerir modificaciones en la interfaz.
- Explicar que el desarrollo de software con técnicas de IA, como la refactorización de código asistida por IA, el desarrollo dirigido por modelos y datos y la optimización de pruebas de software, mejora la eficiencia y la calidad del software.

**Notas para el profesor impartidor. Estas corresponden a la explicación del tema 20 de la semana, la cual debe considerar la realización de ejercicios prácticos durante la sesión.**

Al profesor impartidor, se le recomienda lo siguiente:

- Explicar que la ingeniería de software está en constante evolución debido a avances tecnológicos y cambios en las demandas del mercado.
- Explicar cuáles son las dos tendencias más importantes en la actualidad, mencionando que son la ingeniería de software en la cuarta revolución industrial y en la era de la computación cuántica.
- Explicar que la Cuarta Revolución Industrial, o Industria 4.0, está transformando la forma en que concebimos y desarrollamos el software.
- Explicar que las tecnologías como el Internet de las Cosas (IoT), la Inteligencia Artificial (IA), la robótica avanzada y la computación en la nube están impulsando la transformación en múltiples sectores.
- Explicar que el Internet de las Cosas (IoT) permite la comunicación y el intercambio de datos entre dispositivos interconectados.
- Explicar que IoT tiene aplicaciones en la manufactura inteligente, el hogar inteligente y otros sectores, lo que plantea desafíos en la implementación de esta tecnología en la ingeniería de software.
- Explicar que los ingenieros de software deben abordar estos desafíos y adaptarse a las nuevas demandas tecnológicas para aprovechar al máximo las oportunidades en estas tendencias emergentes.

Recordatorio de la entrega del proyecto, fase 2.

Recordatorio del examen final.

## Anexos

### Rúbrica de la actividad integradora 1

Criterios de evaluación	Nivel de desempeño			%
	Altamente competente 100%-86%	Competente 85%-70%	Aún sin desarrollar la competencia 69%-0%	
1. Desarrolla una visión del producto.	20 - 18	17 - 15	14 - 0	20
	La visión del producto desarrollada se centra en el problema y motiva a todos los involucrados.	La visión del producto desarrollada se centra parcialmente en el problema y motiva parcialmente a todos los involucrados.	La visión del producto desarrollada no se centra en el problema ni motiva a todos los involucrados.	
2. Justifica la metodología seleccionada.	25 - 23	22 - 20	19 - 0	20
	La justificación tiene argumentos válidos para la selección de la metodología.	La justificación tiene parcialmente argumentos para la selección de la metodología.	La justificación no argumenta la selección de la metodología.	
3. Elabora un <i>road map</i> del proyecto.	25 - 23	22 - 20	19 - 0	25
	El road map cuenta con funcionalidades, hitos y fechas de lanzamiento.	El road map cuenta parcialmente con funcionalidades, hitos y fechas de lanzamiento.	El road map cuenta no cuenta con funcionalidades, hitos y fechas de lanzamiento.	
4. Elabora un prototipo con la solución.	15 - 13	12 - 10	9 - 0	35
	El prototipo resuelve las necesidades planteadas en el caso práctico	El prototipo resuelve parcialmente las necesidades planteadas en el caso práctico	El prototipo no resuelve las necesidades planteadas en el caso práctico.	
			<b>TOTAL</b>	<b>100%</b>

## Rúbrica de la actividad integradora 2

Criterios de evaluación	Nivel de desempeño			%
	Altamente competente 100%-86%	Competente 85%-70%	Aún sin desarrollar la competencia 69%-0%	
1. Realiza implementación de seguridad y privacidad.	20 - 18	17 - 15	14 - 0	20
	Incorpora de manera excepcional medidas avanzadas de seguridad y privacidad, con enfoques proactivos y reactivos completos.	Implementa adecuadamente las medidas de seguridad y privacidad, pero con alcance o profundidad limitados.	Falla en implementar medidas básicas de seguridad y privacidad o las implementa de forma errónea.	
2. Incorpora calidad del software.	25 - 23	22 - 20	19 - 0	25
	Aplica modelos de calidad de manera integral y utiliza métricas efectivas para garantizar y mejorar la calidad del software.	Utiliza modelos de calidad y métricas de manera adecuada, aunque no de forma exhaustiva.	No aplica modelos de calidad de manera efectiva o tiene una comprensión limitada de estos.	
3. Realiza ejecución de pruebas.	25 - 23	22 - 20	19 - 0	25
	Realiza pruebas exhaustivas, incluyendo pruebas funcionales y de seguridad, demostrando una alta cobertura y detección de errores.	Ejecuta pruebas con cobertura moderada y detecta fallos importantes, pero podría mejorar en alcance.	Las pruebas son insuficientes o ineficaces, con muchos errores críticos no detectados.	
4. Incorpora gestión y operación de DevOps.	15 - 13	12 - 10	9 - 0	15
	Excelente implementación de CI/CD, uso innovador de contenedores y orquestación, y monitoreo proactivo.	Gestiona bien las operaciones de DevOps, aunque con algunas áreas de mejora en integración y entrega continua.	Gestión de DevOps deficiente, con poca integración y fallos en la entrega y monitoreo continuos.	
5. Realiza despliegue y mantenimiento.	15 - 13	12 - 10	9 - 0	15
	Despliega software de manera eficiente y planifica de forma proactiva el mantenimiento y la gestión de versiones.	Realiza despliegues efectivos y mantiene el software adecuadamente, pero con menor previsión y planificación.	Despliegue y mantenimiento ineficientes, con problemas frecuentes y mala gestión de versiones.	
<b>TOTAL</b>				<b>100%</b>

## Rúbrica de la fase 1 proyecto

Criterios de evaluación	Nivel de desempeño			%
	Altamente competente 100%-86%	Competente 85%-70%	Aún sin desarrollar la competencia 69%-0%	
1. Asegura funcionalidad del software.	20 - 18	17 - 15	14 - 0	20
	El software funciona sin errores y cumple con todos los requisitos especificados, incluyendo los casos de uso avanzados.	El software funciona adecuadamente, pero presenta algunos errores menores; cumple la mayoría de los requisitos.	El software presenta múltiples errores y no cumple con varios requisitos importantes.	
2. Incorpora calidad y eficacia del código.	25 - 23	22 - 20	19 - 0	25
	El código está bien organizado y documentado, fácil de leer y mantener.	El código está razonablemente organizado y tiene una documentación adecuada, con algunos problemas menores de legibilidad o mantenimiento.	El código es desorganizado y carece de documentación, lo que dificulta su comprensión y mantenimiento.	
3. Incorpora innovación y creatividad.	25 - 23	22 - 20	19 - 0	25
	El proyecto muestra una solución original y creativa que destaca por su innovación.	El proyecto es funcional y cumple con el objetivo, pero con poca innovación o creatividad.	El proyecto es convencional sin características innovadoras o creativas.	
4. Mantiene colaboración y trabajo en equipo.	15 - 13	12 - 10	9 - 0	15
	Excelente colaboración entre los miembros del equipo, contribuciones equitativas y comunicación efectiva.	Buena colaboración y comunicación con algunas áreas de mejora.	Poca o ninguna colaboración efectiva; problemas significativos en la comunicación.	
5. Realiza presentación y documentación del proyecto.	15 - 13	12 - 10	9 - 0	15
	La presentación y documentación son completas, claras y bien estructuradas, proporcionando una comprensión profunda del proyecto y sus fases.	La presentación y documentación son adecuadas, con algunos detalles faltantes o áreas poco claras.	La presentación o documentación es insuficiente, desorganizada o incoherente.	
			<b>TOTAL</b>	<b>100%</b>

## Rúbrica de la fase 2 proyecto

Criterios de evaluación	Nivel de desempeño			%
	Altamente competente 100%-86%	Competente 85%-70%	Aún sin desarrollar la competencia 69%-0%	
1. Asegura el cumplimiento de requisitos del software.	20 - 18	17 - 15	14 - 0	20
	El software cumple todos los requisitos especificados y supera las expectativas en funcionalidad y usabilidad.	El software cumple con la mayoría de los requisitos especificados y es funcionalmente adecuado.	El software no cumple con varios requisitos importantes y muestra fallos en funcionalidad.	
2. Incorpora calidad y eficacia del código.	25 - 23	22 - 20	19 - 0	25
	El código es excepcionalmente claro, bien organizado y optimizado. No se detectan errores.	El código es claro y está bien organizado, con pocos errores menores.	El código es difícil de entender y mantener, con errores significativos.	
3. Incorpora innovación y creatividad.	25 - 23	22 - 20	19 - 0	25
	El proyecto muestra una solución innovadora y creativa que se destaca claramente.	El proyecto presenta elementos creativos, pero dentro de un enfoque convencional.	El proyecto carece de originalidad y no ofrece soluciones creativas o nuevas.	
4. Implementación de pruebas y calidad	15 - 13	12 - 10	9 - 0	15
	Se implementan pruebas exhaustivas, incluyendo pruebas unitarias, de integración y de sistema, asegurando alta calidad.	Se realizan pruebas básicas que aseguran la funcionalidad general sin cobertura completa.	Las pruebas son insuficientes o no adecuadas, con muchos problemas de calidad sin resolver.	
5. Realiza documentación y presentación del proyecto.	15 - 13	12 - 10	9 - 0	15
	La documentación es excepcionalmente detallada y clara. La presentación es profesional y bien estructurada.	La documentación es adecuada con algunos detalles faltantes. La presentación es clara.	Documentación pobre o incompleta. Presentación desorganizada o confusa.	
<b>TOTAL</b>				<b>100%</b>

## Prácticas de bienestar

### Práctica 1

<b>Nombre de la práctica</b>	Un momento para respirar.
<b>Descripción de la práctica</b>	Aprender a respirar por la nariz y a tranquilizar tu mente.
<b>Palabras clave</b>	Fortalezas de carácter, autorregulación.
<b>Instrucciones para el aprendizador</b>	<p>La autorregulación, también percibida como control, es una fortaleza de carácter muy importante dentro de la psicología positiva. Este concepto implica regular lo que uno siente y hace, ser disciplinado, así como mantener un control sobre los apetitos y, especialmente, sobre las emociones.</p> <p>En la actualidad vivimos situaciones muy estresantes que provocan que nuestra reacción instintiva y natural ante ellas sea estallar en ira. Pero, las consecuencias de este comportamiento no solo se quedan en nosotros, sino que también pueden llegar a afectar a terceros.</p> <p>A continuación, se presenta un ejercicio que te ayudará a cultivar la fortaleza de autorregulación:</p> <ol style="list-style-type: none"> <li>1. Toma dos minutos de tu tiempo, siéntate en un lugar cómodo, donde no haya mucho ruido que te pueda distraer.</li> <li>2. Escucha música de relajación (crea tu propio ambiente de meditación).</li> <li>3. Comienza a respirar y exhalar por nariz.</li> <li>4. Trata de que tu respiración y exhalación dure el mismo tiempo.</li> <li>5. Fija tu mente en tu respiración, en cómo entra y sale el aire de tu cuerpo.</li> </ol> <p>Así durante dos minutos.</p> <p>Te recomendamos que si durante este periodo algún pensamiento (olvidé algo en la oficina, más tarde tengo que hacer tal actividad, etc.) llega a tu mente, solo déjalo pasar y regresa a la concentración en tu respiración.</p> <p>Al finalizar los dos minutos sentirás paz en tu ser. Comienza a hacer este ejercicio de respiración y meditación todos los días y poco a poco vas aumentando los minutos de este.</p>
<b>Fuente</b>	Conferencia Rosalinda Ballesteros.

### Práctica 2

<b>Nombre de la práctica</b>	Fomentando la atención plena.
<b>Descripción de la práctica</b>	Llevarás a cabo breves ejercicios de meditación para fomentar la atención plena en tus actividades diarias.

<b>Palabras clave</b>	Atención plena, fortalezas de carácter, autorregulación.
<b>Instrucciones para el aprendizador</b>	<p>La meditación es una herramienta que ayuda a mejorar el desempeño de cualquier persona, ya que fomenta el desarrollo de la atención plena en una sola actividad. Para fomentar la atención plena y lograr cada vez más estar en una zona de concentración mientras realizas tus actividades cotidianas, puedes llevar a cabo los siguientes ejercicios de meditación:</p> <p>Encuentra en algún momento del día cinco minutos para ti, siéntate en un lugar cómodo, donde no tengas distracciones.</p> <ol style="list-style-type: none"> <li>1. Haz tres respiraciones profundas por la nariz y exhala por la nariz.</li> <li>2. Comienza a hacer un repaso de tu día, de lo que más te acuerdes, por ejemplo, te levantaste, ¿qué hiciste?, ¿desayunaste?, ¿te bañaste?, ¿diste los buenos días?, etcétera. Si desayunaste, ¿qué fue lo que desayunaste?, ¿te gustó?, ¿tomaste tu alimento despacio o apurado? Si estabas apurado, ¿qué era lo que te tenía en esa situación?</li> <li>3. Sigue meditando en lo que te acuerdes: ¿te molestase con alguien?, ¿por qué?, ¿qué fue lo que pasó?, ¿crees que era posible haber reaccionado de alguna manera más pacífica?</li> </ol> <p>Con este ejercicio te darás cuenta de que reaccionamos o hacemos cosas de manera automática. Algunas veces si estamos más conscientes y presentes, podemos tener otra actitud sin que alguna situación nos afecte demasiado.</p>
<b>Fuente</b>	Eby, D. (s.f.). <i>Creativity and Flow Psychology</i> . Recuperado de <a href="http://talentdevelop.com/articles/Page8.html">http://talentdevelop.com/articles/Page8.html</a>

### Práctica 03

<b>Nombre de la práctica</b>	Experiencias difíciles.
<b>Descripción de la práctica</b>	En esta práctica podrás analizar las estrategias que seguiste para afrontar problemáticas y cómo aprendiste de tales sucesos.
<b>Palabras clave</b>	Resiliencia.
<b>Instrucciones para el aprendizador</b>	<p>Todos hemos pasado por situaciones complejas, no solo en lo laboral, sino también en el ámbito familiar y personal. La manera en que enfrentamos dichos obstáculos es muy diferente, algunas personas continúan con su vida sin problema alguno, a otras tantas se les complica esa transición, también hay quienes no pueden sobreponerse a las experiencias difíciles.</p> <p>La resiliencia es la capacidad de reponerse tras la adversidad, de recuperarse después de vivir experiencias difíciles, dolorosas o traumáticas. Para algunos la resiliencia implica no solo salir adelante después de una situación muy dura, sino incluso crecer o ser mejor a raíz de esta experiencia. (Tarragona, 2012)</p>

	<p>La siguiente práctica te ayudará a fomentar esta importante cualidad:</p> <ol style="list-style-type: none"> <li>1. Crea una tabla con tres columnas y cinco filas.</li> <li>2. En la primera columna escribe un evento difícil o desagradable al que te hayas enfrentado en tu vida.</li> <li>3. En la segunda columna menciona cuáles son tus creencias sobre esa adversidad.</li> <li>4. En la tercera columna describe las consecuencias que tiene esa creencia.</li> <li>5. Cuando termines, lee toda la tabla y reflexiona sobre cómo te ha cambiado cada evento y cómo lo enfrentaste.</li> <li>6. Escribe al final cómo enfrentarías cada evento hoy en día.</li> </ol>
<b>Fuente</b>	<ul style="list-style-type: none"> <li>• Metodología ABC.</li> <li>• Fundamentos de psicología positiva.</li> </ul>

## Práctica 04

<b>Nombre de la práctica</b>	Concentrarse en lo positivo.
<b>Descripción de la práctica</b>	Analizarás sucesos que te hayan ocurrido recientemente, buscando orientar el análisis hacia las consecuencias positivas.
<b>Palabras clave</b>	Resiliencia y esperanza.
<b>Instrucciones para el aprendizador</b>	<p>¿Qué es lo primero que piensas cuando recibes una noticia inesperada?, o bien, ¿qué te imaginas cuando un acontecimiento complejo se presenta ante ti?</p> <p>La mayoría de las personas automáticamente se concentra en el peor de los escenarios independientemente del tipo de noticia que reciban. Martin Seligman sugiere hacer un breve ejercicio para fomentar la resiliencia y la esperanza con base en la premisa antes señalada:</p> <ol style="list-style-type: none"> <li>1. Piensa en una noticia reciente que hayas recibido y que creas que es negativa para ti.</li> <li>2. Luego de analizarla, haz una tabla con tres columnas. En la primera, señala cuál sería el peor de los escenarios posibles que pudieran resultar de esa noticia; en la segunda columna señala cuál sería el mejor de los escenarios posibles, y en la última, cuál es el escenario que realmente tiene mayor probabilidad de ocurrir.</li> <li>3. Reflexiona sobre los tres escenarios, ¿cómo enfrentarías cada uno de ellos?</li> </ol> <p>Procura repetir este ejercicio cada vez que sientas que te enfrentas a una situación complicada. Hacerlo te dará perspectiva y te ayudará a cultivar tu resiliencia.</p>
<b>Fuente</b>	Seligman, M. (2011). <i>Building Resilience</i> . Recuperado de <a href="https://hbr.org/2011/04/building-resilience">https://hbr.org/2011/04/building-resilience</a>

## Práctica 05

<b>Nombre de la práctica</b>	Crecimiento postraumático.
<b>Descripción de la práctica</b>	En esta práctica harás un recuento de las situaciones difíciles a las que te has enfrentado y reflexionarás sobre lo positivo que surgió de ellas.
<b>Palabras clave</b>	Resiliencia.
<b>Instrucciones para el aprendizador</b>	<p>La resiliencia es la capacidad de reponerse tras la adversidad, de recuperarse después de vivir experiencias difíciles, dolorosas o traumáticas. Para algunos la resiliencia implica no solo salir adelante después de una situación muy dura, sino incluso crecer o ser mejor a raíz de esta experiencia. (Tarragona, 2012)</p> <p>La siguiente práctica te ayudará a fomentar esta importante cualidad:</p> <ol style="list-style-type: none"> <li>1. Escribe acerca de un momento en el que enfrentaste una adversidad significativa o pérdida.</li> <li>2. Primero escribe acerca de las puertas que se te cerraron debido a esa adversidad o pérdida, ¿qué perdiste?</li> <li>3. Después escribe acerca de las puertas que se abrieron al término o como secuela de esa adversidad o pérdida.</li> <li>4. ¿Hay nuevas maneras de actuar, pensar o relacionarse que son más probables de suceder ahora?</li> </ol>
<b>Fuente</b>	<ul style="list-style-type: none"> <li>• Ejercicio contribuido por Taylor Kreiss de University of Pennsylvania Positive Psychology Center, y basado en el libro: A Primer in Positive Psychology de Christopher Peterson.</li> </ul>

## Práctica 06

<b>Nombre de la práctica</b>	La mejor versión de ti mismo.
<b>Descripción de la práctica</b>	Escribe acerca de la mejor versión posible de ti mismo durante al menos 20 minutos.
<b>Palabras clave</b>	Emociones positivas, fortalezas de carácter, autorregulación y esperanza.
<b>Instrucciones para el aprendizador</b>	<p>Imagina que dentro de 20 años has crecido en todas las áreas o maneras que te gustaría crecer y las cosas te han salido tan bien como te las imaginaste.</p> <ul style="list-style-type: none"> <li>• ¿Cómo es esa mejor versión de ti mismo?</li> <li>• ¿Qué hace él o ella cotidianamente?</li> <li>• ¿Qué dicen los demás acerca de él o ella?</li> </ul>

	No es necesario que compartas este escrito, ya que el objetivo de esta reflexión es enfocarse en la experiencia que viviste mientras reflexionabas en esa mejor versión posible de ti mismo.
<b>Fuente</b>	<ul style="list-style-type: none"> <li>Ejercicio contribuido por Taylor Kreiss de University of Pennsylvania Positive Psychology Center, y basado en el libro A Primer in Positive Psychology de Christopher Peterson.</li> </ul>

## Práctica 07

<b>Nombre de la práctica</b>	Obtener lo que quieres.
<b>Descripción de la práctica</b>	Reflexionarás sobre alguna meta que desees alcanzar y propondrás una forma de conseguirla.
<b>Palabras clave</b>	Logro, involucramiento, fortalezas de carácter, esperanza, autorregulación, metas y objetivos a largo plazo.
<b>Instrucciones para el aprendizador</b>	<p>Tener una idea clara de lo que desees lograr a corto, mediano y largo plazo es de suma importancia, pues te ayuda a seguir un camino trazado previamente. Para que puedas generar esta guía, responde las siguientes preguntas:</p> <ol style="list-style-type: none"> <li>¿Qué quieres lograr? Al trazar tu meta, procura que esta sea específica, medible, alineada, realista, retadora y con una fecha para lograrla. Piensa en algo y utiliza el método SMART para definirla.</li> <li>¿Qué te impide que lo tengas en este momento?</li> <li>¿Qué sufrimiento estás experimentando en tu vida por no tenerlo en este momento?</li> <li>¿Qué placer, involucramiento, relación, significado o logro tendrías en tu vida si tuvieras eso en este momento?</li> <li>¿Qué hábitos te detienen o no te dejan avanzar hacia eso que quieres?</li> <li>¿Qué nuevos hábitos podrías generar para ayudarte a obtener lo que quieres?</li> <li>¿Qué dos cosas podrías hacer para romper con los hábitos que no te permiten avanzar hacia lo que quieres y generar hábitos nuevos?</li> <li>¿Te comprometes a hacer esas dos cosas? Si es así, ¿cuándo las harás?</li> </ol> <p>Escribe tus resultados en un sitio donde puedas verlos constantemente.</p>
<b>Fuente</b>	<ul style="list-style-type: none"> <li>Ejercicio contribuido por Taylor Kreiss de University of Pennsylvania Positive Psychology Center, y basado en el libro A Primer in Positive Psychology de Christopher Peterson.</li> </ul>

## Práctica 08

<b>Nombre de la práctica</b>	Felicidad en el trabajo.
<b>Descripción de la práctica</b>	Reflexionarás sobre las distintas dimensiones de tu vida cotidiana, enfocando el análisis a cómo fomentar un estado de ánimo y relaciones positivos en el ámbito laboral.

Palabras clave	Involucramiento, emociones positivas, relaciones positivas.
Instrucciones para el aprendizador	<p>Elegir conscientemente maneras de incrementar la felicidad en el trabajo puede hacer la diferencia en cómo nosotros nos sentimos y qué tan bien nos desempeñamos. En lugar de quejarnos del trabajo, ¿por qué no pensar en cómo podemos obtener mayor felicidad de lo que hacemos?</p> <p>Estar más involucrados en lo que hacemos contribuye a nuestra felicidad y bienestar, y nos lleva a un mejor desempeño y productividad. A manera de reflexión, responde las siguientes preguntas que están enfocadas en distintas dimensiones de tu vida:</p> <ul style="list-style-type: none"> <li>• <b>Dar:</b> ¿cómo estoy apoyando a mis colaboradores, compañeros, líderes, proveedores y clientes?</li> <li>• <b>Relaciones:</b> ¿cómo puedo mejorar mis relaciones en el trabajo?, ¿cómo logro un balance entre la vida laboral y familiar?</li> <li>• <b>Ejercicio:</b> ¿cómo puedo integrar la actividad física dentro de mis actividades diarias?, ¿cómo aseguro que estoy comiendo bien y descansando lo suficiente?</li> <li>• <b>Conciencia:</b> ¿cómo puedo construir momentos de atención plena en mi día laboral?</li> <li>• <b>Ensayo:</b> ¿qué habilidades estoy construyendo?, ¿qué cosas nuevas he experimentado?</li> <li>• <b>Dirección:</b> ¿cuáles son mis metas laborales hoy, esta semana, este año?, ¿cómo caben y contribuyen estas con mis metas de vida y me ayudan a desarrollar mis competencias en la construcción de mis relaciones y cómo contribuyo con lo anterior a ayudar a otros?, ¿cómo se pueden alinear mis metas laborales con las de mi equipo y la organización?</li> <li>• <b>Resiliencia:</b> ¿cuáles son mis tácticas para lidiar con los retos difíciles en el trabajo?, ¿me estoy enfocando en lo que puedo controlar?, ¿necesito pedir ayuda a otros?, ¿hay alguien a mi alrededor que requiere de mi ayuda?</li> <li>• <b>Emoción:</b> ¿qué cosas, aunque sean pequeñas, puedo encontrar que me pueden hacer sentir bien en mi trabajo hoy?, ¿qué me ha hecho sonreír?</li> </ul>
Fuente	Tomado del Catálogo de actividades para profesores.

## Práctica 9

Nombre de la práctica	Interacciones positivas.
Descripción de la práctica	Reflexionarás sobre las cualidades positivas que aprecias de las personas con las que interactúas diariamente.
Palabras clave	Relaciones positivas.
Instrucciones para el aprendizador	Puedes obtener mayor gozo de los momentos que compartes con tus colegas si te tomas el tiempo para pensar en lo que valoras y aprecias de

ellos. Diversas investigaciones muestran que enfocarse en lo positivo que sucede diariamente ayuda a incrementar nuestra felicidad y lo mismo aplica a todas nuestras relaciones cercanas.

El psicólogo John Gottman sugiere que, para tener relaciones felices con alguna persona, es necesario aspirar a tener cinco interacciones positivas por cada interacción negativa que se tenga con ella. Enfócate en tus compañeros y/o colegas y piensa en las siguientes preguntas. En cada caso, anota ejemplos específicos.

1. ¿Qué te atrajo de tus compañeros cuando se conocieron?
2. ¿Qué cosas han disfrutado al hacerlas juntos?
3. ¿Qué cosas realmente aprecias de ellos en este momento?
4. ¿Cuáles son sus fortalezas?

Ahora, lo más importante es que cuando estés con tus compañeros te tomes el tiempo para darte cuenta y reconocer estas cualidades, sus fortalezas y las cosas que ellos hacen que realmente aprecies, así como los momentos agradables que han compartido.

Piensa en estas declaraciones:

- “Realmente me encanta cuando ellos...”.
- “Son tan buenos para...”.
- “Viéndolos hacer..., me recuerda ese fantástico día cuando nosotros...”.

Aunque realizar dicho análisis con todas las personas que conoces resulta poco práctico, puedes usar los mismos principios para mejorar tus relaciones en general. Por ejemplo, antes de pasar tiempo con alguien tómate un momento para pensar en aquellas cosas que te gustan, aprecias o admiras de esa persona o cómo te hacen sentir bien. Asimismo, después de pasar tiempo con esa persona, piensa en las cosas que apreciaste o lo que disfrutaste del tiempo que pasaron juntos.

**Fuente**

Basado en el Catálogo de actividades para profesores.

**Práctica 10**

<b>Nombre de la práctica</b>	Las fortalezas se muestran en nuestras historias.
<b>Descripción de la práctica</b>	Reflexionarás sobre las fortalezas de carácter que aplicaste en una situación.
<b>Palabras clave</b>	Fortalezas de carácter.
<b>Instrucciones para el aprendizador</b>	<p>Antes de comenzar el ejercicio, ¿sabes cuáles son las fortalezas de carácter? Consulta la descripción de las 24 fortalezas de carácter en la siguiente liga:</p> <p><b>El siguiente enlace es externo a la Universidad Tecmilenio, al acceder a este considera que debes apegarte a sus términos y condiciones.</b></p> <p><a href="http://www.viacharacter.org/www/Character-Strengths/VIA-Classification">http://www.viacharacter.org/www/Character-Strengths/VIA-Classification</a></p>

	<p>Luego de que leas cuáles son las fortalezas de carácter, realiza lo que se pide a continuación:</p> <ol style="list-style-type: none"> <li>1. Describe detalladamente, mediante un texto, una anécdota en la que hayas llevado a cabo alguna acción de la mejor manera posible, o bien, que hayas actuado por encima de lo ordinario. Procura enfocarlo al entorno laboral.</li> <li>2. Puede ser cualquier suceso que te haya marcado por la manera en que te desarrollaste.</li> <li>3. Señala en tu descripción: ¿qué ocurrió?, ¿qué papel jugaste en el suceso?, ¿qué acciones llevaste a cabo que fueron de utilidad para ti y para los demás?</li> <li>4. Luego de que hayas terminado de escribir, lee tu texto y subraya las palabras y oraciones que te den una idea sobre cómo usaste cualquiera de las 24 fortalezas de carácter.</li> <li>5. Observa y clasifica cuáles son las fortalezas que usaste en tu anécdota. Reflexiona sobre el impacto que estas pueden tener en tu desempeño cotidiano.</li> </ol>
<b>Fuente</b>	<p>Niemiec, R. (2016). <i>How to Assess Your Strengths: 5 Tactics for Self-Growth</i>. Recuperado de <a href="https://www.psychologytoday.com/us/blog/what-matters-most/201603/how-assess-your-strengths-5-tactics-self-growth">https://www.psychologytoday.com/us/blog/what-matters-most/201603/how-assess-your-strengths-5-tactics-self-growth</a></p>

## Práctica 11

<b>Nombre de la práctica</b>	Tus fortalezas en los ojos del otro.
<b>Descripción de la práctica</b>	En la práctica podrás reflexionar sobre la percepción que otros tienen sobre tus fortalezas de carácter.
<b>Palabras clave</b>	Fortalezas de carácter.
<b>Instrucciones para el aprendiz</b>	<p>¿Recuerdas alguna ocasión en la que hablaste con algún colega y este te reveló algo positivo que piensa de ti? Cuando esto ocurre, usualmente deja huella en nuestros comportamientos y acciones, pues nos damos cuenta de que las personas tienen percepciones sobre nuestras fortalezas que nosotros mismos no vislumbramos. Haz lo siguiente:</p> <ol style="list-style-type: none"> <li>1. Piensa sobre alguna vez que algún compañero de trabajo te compartió lo que piensa de ti y que te haya sorprendido.</li> <li>2. Piensa en lo siguiente: ¿qué fue lo que te llamó más la atención?, ¿qué fortalezas vio en ti que pensaste que no tenías tan desarrolladas?</li> <li>3. Por último, señala en un texto por qué consideras que esta revelación te causó tanto impacto, así como la manera en que te ayudó a cultivar tus fortalezas de carácter.</li> </ol>
<b>Fuente</b>	<p>Niemiec, R. (2016). <i>How to Assess Your Strengths: 5 Tactics for Self-Growth</i>. Recuperado de <a href="https://www.psychologytoday.com/us/blog/what-matters-most/201603/how-assess-your-strengths-5-tactics-self-growth">https://www.psychologytoday.com/us/blog/what-matters-most/201603/how-assess-your-strengths-5-tactics-self-growth</a></p>

## Práctica 12

Nombre de la práctica	Plantea tus objetivos como metas de aproximación y replantea tus metas de evitación.
Descripción de la práctica	Con base en lo que plantea Grenville (2012), en la práctica podrás definir diferentes tipos de metas y encontrar la mejor manera de conseguirlas.
Palabras clave	Objetivos, metas y planes.
Instrucciones para el aprendizador	<p>La autora Bridget Grenville-Cleave (2012) comenta que en el establecimiento de metas es importante distinguir los tipos de metas que hay y menciona dos:</p> <ol style="list-style-type: none"> <li>1. Metas de aproximación (<i>approach</i>): son las metas con resultados positivos (deseables, placenteros, benéficos o que nos gustaría tener) y hacia las cuales trabajamos.</li> <li>2. Metas de evitación (<i>avoidance</i>): son las metas con resultados negativos (indeseables, dolorosos, dañinos, o nos disgustan) y en las cuales trabajamos para evitarlas.</li> </ol> <p>Ejemplo:</p> <p><b>Meta de aproximación:</b></p> <ul style="list-style-type: none"> <li>• Ser más eficiente.</li> <li>• Ser amigable y extrovertido en reuniones.</li> <li>• Asumir el rol de líder en el trabajo.</li> </ul> <p><b>Meta de evitación:</b></p> <ul style="list-style-type: none"> <li>• Dejar de aplazar.</li> <li>• Dejar de ser tan tímido en las reuniones.</li> <li>• No pasar desapercibido en el trabajo.</li> </ul> <p>Las investigaciones que se han realizado respecto a estos tipos de metas muestran que perseguir metas de evitación resulta en un detrimento del bienestar. Estos descubrimientos sugieren que el establecer metas de aproximación o replantear las metas de evitación es benéfico.</p> <p>Reflexiona lo siguiente:</p> <ul style="list-style-type: none"> <li>• ¿Qué tipo de metas te has planteado tú?</li> <li>• ¿Hay algunas metas que puedas replantear en una forma más positiva?</li> <li>• ¿Cuándo las tendrás listas?</li> </ul>
Fuente	Grenville, B. (2012). <i>GOAL-SETTING SECRETS</i> . Recuperado de <a href="http://positivepsychologynews.com/news/bridget-grenville-cleave/2012013120696">http://positivepsychologynews.com/news/bridget-grenville-cleave/2012013120696</a>