

## SÍLABO

### Curso integrador II: software (100000S12F)

### 2025 - Ciclo 1 Marzo

#### 1. DATOS GENERALES

1.1. Carrera:	Ingeniería de Software
1.2. Créditos:	3
1.3. Enseñanza de curso:	Presencial
1.4. Horas semanales:	4

#### 2. FUNDAMENTACIÓN

El presente curso tiene como propósito que el estudiante integre las competencias adquiridas a lo largo de la carrera. A partir de problemas complejos o mejoras reforzará la capacidad de análisis para encontrar una solución tecnológica al problema establecido a fin de desarrollar un software de alta calidad. Este curso le permitirá cimentar e integrar las competencias adquiridas que luego serán usadas en el ejercicio profesional del estudiante en las áreas de TI de cualquier organización.

#### 3. SUMILLA

El curso es de naturaleza teórico - práctica que le permitirá al estudiante conocer las herramientas de programación, metodologías para el desarrollo de software (Devops, SRE) y la gestión de proyectos (Scrum, PMBOK), para desarrollar un aplicativo informático que resuelva un problema real del entorno y satisfacer una necesidad de las empresas o negocios.

#### 4. LOGRO GENERAL DE APRENDIZAJE

Al finalizar el curso, el estudiante desarrolla un software acorde con el modelo de desarrollo establecido, considerando las normas de calidad de software vigente.

#### 5. UNIDADES Y LOGROS ESPECÍFICOS DE APRENDIZAJE

<b>Unidad de aprendizaje 1:</b> Identificación y Planificación de software..	Semana 1 y 2
<b>Logro específico de aprendizaje:</b> Al finalizar la unidad, el estudiante planifica el proyecto de desarrollo de software, tomando en cuenta la necesidad del negocio o la empresa.	
<b>Temario:</b> <ul style="list-style-type: none"><li>• Presentación del curso. Explicación de los tipos de proyectos a presentar. Esquema de trabajo de perfil del proyecto.</li><li>• Lenguajes de programación de alto nivel.</li></ul>	
<b>Unidad de aprendizaje 2:</b> Análisis de software.	Semana 3 y 4
<b>Logro específico de aprendizaje:</b> Al finalizar la unidad, el estudiante analiza las funcionalidades del software aplicando la metodología de gestión de proyectos.	
<b>Temario:</b> <ul style="list-style-type: none"><li>• Taller de Desarrollo Ágil: Scrum. Definir la metodología para gestionar el proyecto: basado en SCRUM, PMBOK e ITIL. Asesoría del docente con relación análisis de los sistemas a desarrollar (Día 1).</li><li>• Taller de desarrollo de una aplicación. Asesoría del docente con relación análisis de los sistemas a desarrollar (Día 2).</li></ul>	

<b>Unidad de aprendizaje 3:</b> Diseño de la solución.	Semana 5,6,7 y 8
<b>Logro específico de aprendizaje:</b> Al finalizar la unidad, el estudiante diseña el software considerando las metodologías y funcionalidades que requiere el negocio.	
<b>Temario:</b> <ul style="list-style-type: none"> <li>• Metodología y técnicas Devops. Metodología y técnicas SRE.</li> <li>• Marco de trabajo SCRUM. Selección de arquitectura de software.</li> <li>• Asesoría del docente relación del documento al diseño de los sistemas a desarrollar (Día 1).</li> <li>• Asesoría del docente relación al diseño de los sistemas a desarrollar (Día 2).</li> </ul>	
<b>Unidad de aprendizaje 4:</b> Desarrollo de los componentes del software.	Semana 9,10,11,12 y 13
<b>Logro específico de aprendizaje:</b> Al finalizar la unidad, el estudiante desarrolla el prototipo de las partes del software, considerando las normas de calidad de software vigente.	
<b>Temario:</b> <ul style="list-style-type: none"> <li>• Selección del patrón de diseño.</li> <li>• Desarrollo de los componentes y reutilización.</li> <li>• Desarrollo de despliegue de software.</li> <li>• Asesoría del docente relación desarrollo de los componentes del sistema a desarrollar.</li> </ul>	
<b>Unidad de aprendizaje 5:</b> Probar el software desarrollado.	Semana 14 y 15
<b>Logro específico de aprendizaje:</b> Al finalizar la unidad, el estudiante prueba las funcionalidades del prototipo del software considerando los requisitos establecidos por el negocio.	
<b>Temario:</b> <ul style="list-style-type: none"> <li>• Plan de pruebas de SW. Ejecución de Pruebas orientadas a seguridad: ataques, inyecciones SQL.</li> <li>• Ejecución de Pruebas orientadas a usabilidad. Validación y verificación de requerimientos de SW.</li> </ul>	
<b>Unidad de aprendizaje 6:</b> Integrar las funcionalidades del software, considerando su documentación.	Semana 16,17 y 18
<b>Logro específico de aprendizaje:</b> Al finalizar la unidad, el estudiante integra las funcionalidades y documentación del software desarrollado de acuerdo con las normas de calidad del software vigente.	
<b>Temario:</b> <ul style="list-style-type: none"> <li>• Elaboración de Documentación de Proyecto: Manuales (de usuario y técnico), Informe del Proyecto, Resumen Ejecutivo.</li> <li>• Presentación y Exposición Final del Proyecto. Exposiciones y sustentaciones de proyectos.</li> <li>• Revisión final del Proyecto.</li> </ul>	

## 6. METODOLOGÍA

Para el desarrollo de los aprendizajes del curso, una de las estrategias que se propone es la exposición del docente que proporciona la construcción de los conocimientos a partir de ejemplos y casuísticas que faciliten la comprensión. Asimismo, se promueve la participación activa y permanente del estudiante a través del desarrollo de ejercicios, lecturas, absolución de preguntas, en forma individual y grupal (aprendizaje colaborativo) lo que permite un trabajo metacognitivo, a través de la actividad autónoma del estudiante en el desarrollo de las evaluaciones del curso (aprendizaje autónomo). Por ello es importante que el estudiante asista a las clases, habiendo leído los temas correspondientes a cada sesión. Finalmente, se utilizan otros recursos, como: pizarra, multimedia, videos (aprendizaje para la era digital) y comunicación a través de medios complementarios como correos electrónicos para fomentar una mayor interacción con el estudiante.

## 7. SISTEMA DE EVALUACIÓN

El cálculo del promedio final se hará de la siguiente manera:

$$(15\%)APF1 + (15\%)APF2 + (20\%)APF3 + (20\%)APF4 + (30\%)PROY$$

Donde:

Tipo	Descripción	Semana	Observación
APF1	AVANCE DE PROYECTO FINAL 1	4	Presentación Grupal
APF2	AVANCE DE PROYECTO FINAL 2	8	Presentación Grupal
APF3	AVANCE DE PROYECTO FINAL 3	13	Presentación Grupal
APF4	AVANCE DE PROYECTO FINAL 4	15	Presentación Grupal
PROY	PROYECTO FINAL	18	Presentación Grupal Proyecto Final

#### Indicaciones sobre Fórmulas de Evaluación:

1. La nota mínima aprobatoria final es de 12.
2. En este curso, no aplica examen rezagado.
3. En este curso, ninguna nota se reemplaza.

## 8. FUENTES DE INFORMACIÓN

#### Bibliografía Base:

- Luis, C., Perla, V., & Humberto, C. . *Arquitectura del Software*. Cengage Learning Editores SA. <https://tubiblioteca.utp.edu.pe/cgi-bin/koha/opac-detail.pl?biblionumber=29509>

#### Bibliografía Complementaria:

- Monte Galiano, Josep. (2016). *Implantar scrum con éxito*. Editorial UOC. <https://tubiblioteca.utp.edu.pe/cgi-bin/koha/opac-detail.pl?biblionumber=36942>
- Ivana Harari ; Díaz, Javier ; Ana Paola Amadeo. (2013). *Guía de recomendaciones para diseño de software centrado en el usuario*. D - Editorial de la Universidad Nacional de La Plata. <https://tubiblioteca.utp.edu.pe/cgi-bin/koha/opac-detail.pl?biblionumber=36892>
- Robaina, Daniel Alfonso - Director; Malleuve Martínez, Annette. *Modelo de dirección estratégica con enfoque de arquitectura empresarial: para la integración del sistema de dirección de la empresa*. Editorial Universitaria. <https://tubiblioteca.utp.edu.pe/cgi-bin/koha/opac-detail.pl?biblionumber=37464>

## 9. COMPETENCIAS

Carrera	Competencias específicas
Ingeniería de Software	<ul style="list-style-type: none"> <li>• Desarrollo de Software</li> <li>• Análisis de sistemas</li> <li>• Soluciones informáticas</li> </ul>
	<ul style="list-style-type: none"> <li>• Análisis de sistemas</li> <li>• Soluciones informáticas</li> <li>• Desarrollo de software</li> </ul>

## 10. CRONOGRAMA DE ACTIVIDADES

Unidad de aprendizaje	Semana	Sesión	Tema	Actividades y evaluaciones
		1	Presentación del curso. Explicación de los tipos de proyectos a presentar. Esquema de trabajo de perfil del proyecto.	<ul style="list-style-type: none"> <li>• Ejecución de prueba de entrada de no más de 45 minutos que evalúen los saberes previos de los alumnos. Dinámica de Trabajo en Equipo: Juego de Roles. Tormenta de ideas de proyectos potenciales.</li> </ul>

<b>Unidad 1</b> Identificación y Planificación de software.	1			Mediante un borrador cada alumno identifica las características de un proyecto
		2	Presentación del curso. Explicación de los tipos de proyectos a presentar. Esquema de trabajo de perfil del proyecto.	<ul style="list-style-type: none"> <li>Ejecución de prueba de entrada de no más de 45 minutos que evalúen los saberes previos de los alumnos. Dinámica de Trabajo en Equipo: Juego de Roles. Tormenta de ideas de proyectos potenciales. Mediante un borrador cada alumno identifica las características de un proyecto</li> </ul>
	2	3	Lenguajes de programación de alto nivel.	<ul style="list-style-type: none"> <li>Herramientas usadas en la elaboración de proyectos y metodologías.</li> <li>Estructura del proyecto final.</li> </ul>
		4	Lenguajes de programación de alto nivel.	<ul style="list-style-type: none"> <li>Herramientas usadas en la elaboración de proyectos y metodologías.</li> <li>Estructura del proyecto final.</li> </ul>
<b>Unidad 2</b> Análisis de software	3	5	Taller de Desarrollo Ágil: Scrum. Definir la metodología para gestionar el proyecto: basado en SCRUM, PMBOK e ITIL. Asesoría del docente con relación análisis de los sistemas a desarrollar (Día 1).	<ul style="list-style-type: none"> <li>Seminario de Desarrollo Ágil: SCRUM Realizar seminario – Taller para definir la metodología para gestionar el proyecto. Análisis de requerimiento del negocio.</li> </ul>
		6	Taller de Desarrollo Ágil: Scrum. Definir la metodología para gestionar el proyecto: basado en SCRUM, PMBOK e ITIL. Asesoría del docente con relación análisis de los sistemas a desarrollar (Día 1).	<ul style="list-style-type: none"> <li>Seminario de Desarrollo Ágil: SCRUM Realizar seminario – Taller para definir la metodología para gestionar el proyecto. Análisis de requerimiento del negocio.</li> </ul>
			Taller de desarrollo de una aplicación. Asesoría del docente con relación análisis de los sistemas a desarrollar	<ul style="list-style-type: none"> <li>Presentación del Informe de Proyecto.</li> </ul>

	4	7	(Día 2).	Entregable 1 al 20%: Perfil del proyecto y análisis de requerimiento.
		8	Evaluación	• AVANCE DE PROYECTO FINAL 1
<b>Unidad 3</b> Diseño de la solución	5	9	Metodología y técnicas Devops. Metodología y técnicas SRE.	• Aplicación de Devops y SRE. Revisión de avance del trabajo.
		10	Metodología y técnicas Devops. Metodología y técnicas SRE.	• Aplicación de Devops y SRE. Revisión de avance del trabajo.
	6	11	Marco de trabajo SCRUM. Selección de arquitectura de software.	• Elaboración de historias de usuario. Planificación de los sprint. Eventos de los sprint. Definir la arquitectura de software.
		12	Marco de trabajo SCRUM. Selección de arquitectura de software.	• Elaboración de historias de usuario. Planificación de los sprint. Eventos de los sprint. Definir la arquitectura de software.
	7	13	Asesoría del docente relación del documento al diseño de los sistemas a desarrollar (Día 1).	• Elaboración de los cronogramas de los sprint y burndown chart. Revisión de avance del trabajo.
		14	Asesoría del docente relación del documento al diseño de los sistemas a desarrollar (Día 1).	• Elaboración de los cronogramas de los sprint y burndown chart. Revisión de avance del trabajo.
	8	15	Asesoría del docente relación al diseño de los sistemas a desarrollar (Día 2).	• Entregable 2 del proyecto al 40%: APF2 Exposición del diseño detallado.
		16	Evaluación	• AVANCE DE PROYECTO FINAL 2
		17	Selección del patrón de diseño.	• Elaborar diagrama de base de datos e interfaz gráfica

**Unidad 4**  
Desarrollo de los  
componentes del  
software

9	18	Selección del patrón de diseño.	<ul style="list-style-type: none"> <li>• Elaborar diagrama de base de datos e interfaz gráfica</li> </ul>
10	19	Desarrollo de los componentes y reutilización.	<ul style="list-style-type: none"> <li>• Identificación de los componentes de software y reutilización Ventajas y desventajas de la reutilización de software. Ejemplos de reutilización de software. Revisión de avance del trabajo.</li> </ul>
	20	Desarrollo de los componentes y reutilización.	<ul style="list-style-type: none"> <li>• Identificación de los componentes de software y reutilización Ventajas y desventajas de la reutilización de software. Ejemplos de reutilización de software. Revisión de avance del trabajo.</li> </ul>
11	21	Desarrollo de despliegue de software.	<ul style="list-style-type: none"> <li>• Métodos y procesos de despliegue de software. Revisión de avance del trabajo.</li> </ul>
	22	Desarrollo de despliegue de software.	<ul style="list-style-type: none"> <li>• Métodos y procesos de despliegue de software. Revisión de avance del trabajo.</li> </ul>
12	23	Asesoría del docente relación desarrollo de los componentes del sistema a desarrollar.	<ul style="list-style-type: none"> <li>• Revisión de avance del trabajo.</li> </ul>
	24	Asesoría del docente relación desarrollo de los componentes del sistema a desarrollar.	<ul style="list-style-type: none"> <li>• Revisión de avance del trabajo.</li> </ul>
13	25	Asesoría del docente relación desarrollo de los componentes del sistema a desarrollar.	<ul style="list-style-type: none"> <li>• Entregable 3: Avance del proyecto al 80%.Exposición del informe de los componentes del sistema detallado. Informe de los componentes detallado del sistema a desarrollar de la organización.</li> </ul>

		26	Evaluación	<ul style="list-style-type: none"> <li>• AVANCE DE PROYECTO FINAL 3</li> </ul>
<b>Unidad 5</b> Probar el software desarrollado	14	27	Plan de pruebas de SW. Ejecución de Pruebas orientadas a seguridad: ataques, inyecciones SQL.	<ul style="list-style-type: none"> <li>• Elaboración de Plan de Pruebas</li> <li>Revisión de avance del trabajo.</li> </ul>
		28	Plan de pruebas de SW. Ejecución de Pruebas orientadas a seguridad: ataques, inyecciones SQL.	<ul style="list-style-type: none"> <li>• Completado (Desarrollo de la aplicación terminada en versión beta y Ejecución de Pruebas de Software).</li> </ul>
	15	29	Ejecución de Pruebas orientadas a usabilidad. Validación y verificación de requerimientos de SW.	<ul style="list-style-type: none"> <li>• Exposición de los temas de clase</li> </ul>
		30	Evaluación	<ul style="list-style-type: none"> <li>• AVANCE DE PROYECTO FINAL 4</li> </ul>
<b>Unidad 6</b> Integrar las funcionalidades del software, considerando su documentación	16	31	Elaboración de Documentación de Proyecto: Manuales (de usuario y técnico), Informe del Proyecto, Resumen Ejecutivo.	<ul style="list-style-type: none"> <li>• Elaboración de Documentación de Proyecto: Manuales, Informe del Proyecto, Resumen Ejecutivo. Revisión de avance del trabajo.</li> </ul>
		32	Elaboración de Documentación de Proyecto: Manuales (de usuario y técnico), Informe del Proyecto, Resumen Ejecutivo.	<ul style="list-style-type: none"> <li>• Elaboración de Documentación de Proyecto: Manuales, Informe del Proyecto, Resumen Ejecutivo. Revisión de avance del trabajo.</li> </ul>
	17	33	Presentación y Exposición Final del Proyecto. Exposiciones y sustentaciones de proyectos.	<ul style="list-style-type: none"> <li>• Presentaciones, Exposiciones. Trabajo con un avance al 100%. Presentación y Exposición Final del Proyecto.</li> </ul>
		34	Presentación y Exposición Final del Proyecto. Exposiciones y sustentaciones de proyectos.	<ul style="list-style-type: none"> <li>• Presentaciones, Exposiciones. Trabajo con un avance al 100%. Presentación y Exposición Final del Proyecto.</li> </ul>
			Revisión final del Proyecto.	<ul style="list-style-type: none"> <li>• Observaciones y recomendaciones del proyecto.</li> </ul>

	18	35	Evaluación	• PROYECTO FINAL
--	----	----	------------	------------------