

PROGRAMA DE CURSO

PROYECTO DE SOFTWARE

A. Antecedentes generales del curso:

Departamento	Departamento de Ciencias de la Computación				
Nombre del curso	Proyecto de software	Código	CC5402	Créditos	12
Nombre del curso en inglés	Software Project				
Carácter del curso	Obligatorio	X			
Requisitos	CC5205, CC4402				

B. Descripción del curso

El curso Proyecto de Software es un curso integrador de todos los aprendizajes adquiridos a lo largo de la carrera de Ingeniería Civil en Computación (*capstone course*). Su propósito es la realización de un proyecto, auténtico reflejo de la práctica profesional que desempeñarán los y las estudiantes al titularse. Este trabajo se realiza mediante el desarrollo de un proyecto basado en software para una organización externa al departamento de computación.

El curso sigue un enfoque de vinculación con el medio guiado por proyectos basado fundamentalmente en metodologías ágiles de desarrollo de software. Se sigue una estrategia incremental organizada en tres iteraciones. El desarrollo del proyecto de software se realiza en equipos interactuando sincrónicamente de forma virtual.

El curso comprende además una serie de actividades sincrónicas principalmente relativas a los aspectos docentes del curso:

Formato presencial:

- Dos semanas introductorias dedicadas a presentar a los estudiantes tanto los proyectos como las metodologías a aplicar.
- Una presentación pública al final de cada iteración donde se expone el trabajo realizado hasta el momento, la planificación, los desafíos enfrentados y las reflexiones del trabajo realizado.

Formato Virtual:

- Reuniones semanales de coordinación, reflexión y guía con el cuerpo docente.

Modalidad de trabajo:

La modalidad de trabajo es híbrida. Los y las estudiantes deben considerar 16 horas de trabajo asincrónico del equipo completo en el desarrollo del proyecto de software (esto considera desarrollo del software y reuniones con los clientes). Asimismo, se consideran dos horas semanales de reuniones sincrónicas con el cuerpo docente, para retroalimentar a los equipos de trabajo, tomando en cuenta sus avances.

Los equipos se organizan en torno a horarios de trabajo, considerando el contacto con clientes quienes determinan que es necesario o conveniente resolver cierto problema mediante un proyecto. Estos actores son quienes presentan los proyectos y van interactuando activamente con los/las estudiantes para mantener alineados el problema y la solución que se va desarrollando. También son fundamentales los/las usuarios/as, que no necesariamente son los clientes, donde serán los/las usuarios /as, quienes usarán el software desarrollado.

Los y las estudiantes deberán interactuar con las/los clientes a partir de tener un producto parcialmente construido para validar la potencialidad de adopción efectiva. Cabe destacar que los/las usuarios/as pueden ser los propios clientes, aunque por lo general no lo son.

El curso tributa a las siguientes competencias específicas (CE) y genéricas (CG):

CE1: Analizar problemas computacionales, construir modelos, expresándolos en representaciones y lenguajes formales adecuados.

CE5: Concebir, diseñar y construir soluciones de software, siguiendo un proceso sistemático y cuantificable, acorde a los fundamentos y eligiendo el paradigma y las técnicas más adecuadas.

CE6: Desarrollar software en una amplia variedad de plataformas y lenguajes de programación.

CE7: Gestionar proyectos de diseño, desarrollo, implementación y evolución de soluciones de software, contemplando tanto los procesos involucrados como el producto esperado, su calidad y respuesta efectiva al problema que aborda.

CE9: Desarrollar soluciones computacionales de manera interdisciplinaria y colaborativa.

CG1: Comunicación académica y profesional

Comunicar en español de forma estratégica, clara y eficaz, tanto en modalidad oral como escrita, puntos de vista, propuestas de proyectos y resultados de investigación fundamentados, en situaciones de comunicación compleja, en ambientes sociales, académicos y profesionales.

CG3: Compromiso ético

Actuar de manera responsable y honesta, dando cuenta en forma crítica de sus propias acciones y sus consecuencias, en el marco del respeto hacia la dignidad de las personas

y el cuidado del medio social, cultural y natural.

CG4: Trabajo en equipo

Trabajar en equipo, de forma estratégica y colaborativa, en diversas actividades formativas, a partir de la autogestión de sí mismo y de la relación con el otro, interactuando con los demás en el ámbito de gestión de proyectos de software.

CG5: Innovación

Concebir ideas viables y novedosas que generen valor para resolver necesidades latentes, materializadas en productos, servicios o en mejoras a procesos dentro de un sistema u organización, considerando el contexto sociocultural y económico y los beneficios para el usuario.

C. Resultados de aprendizaje:

1. Planifica, define y negocia alcances del proyecto, considerando desafíos técnicos, incertidumbres organizacionales y evolución del foco del proyecto.
2. Trabaja en equipo para desarrollar soluciones de software en diversos dominios de aplicación, considerando la autogestión para lograr el objetivo.
3. Analiza, diseña y construye soluciones innovadoras de software de alta calidad técnica, para resolver un problema real de una organización, considerando potenciales impactos éticos y ambientales.
4. Presenta y describe con claridad y de forma estructurada las distintas facetas del proyecto: organización, problema, solución, demo del producto desarrollado y reflexiona sobre desafíos enfrentados.

D. Planificación y cronograma del curso:

Semana	Actividades		Evaluación	Resultado de Aprendizaje evaluado
	Sincrónicas / Presenciales	Asincrónicas / No presenciales		
1	Presentación del curso Distribución de proyectos y tutores	Organización de los horarios de trabajo del equipo Primer contacto con los clientes		-
2	- Presentación de los objetivos del curso: objetivos docentes y de impacto en la organización cliente - Presentación 0: los alumnos exponen ante sus compañeros la descripción general de su proyecto	Preparación de la presentación 0	Evaluación formativa relativa a: - claridad para describir el alcance del proyecto - calidad de la estructura y producción de la presentación	RA1, RA4

3, 4, 5, 6.a	- Reunión virtual semanal con el cuerpo docente - Reunión con cliente (es opcional que sea presencial o virtual)	Desarrollo del proyecto Reuniones planificadas y no planificadas con los clientes	En cada reunión semanal se hace una reflexión formativa sobre: - trabajo realizado - planificación de los siguientes pasos - desafíos enfrentados tanto técnicos como de trabajo en equipo e impacto ético del proyecto	RA1, RA2, RA3
6.b	Presentación 1: - descripción de la organización cliente - problema abordado - trabajo realizado hasta el momento - planificación - reflexión sobre la experiencia	Preparación de la presentación 1	Evaluación sumativa con las ponderaciones correspondientes a la iteración 1 (25% de la calificación final)	RA1, RA2, RA3, RA4

7, 8, 9, 10.a	- Reunión virtual semanal con el cuerpo docente - Reunión con cliente (es opcional que sea presencial o virtual)	Desarrollo del proyecto Reuniones planificadas y no planificadas con los clientes	En cada reunión semanal se hace una reflexión formativa sobre: - trabajo realizado - planificación de los siguientes pasos - desafíos enfrentados tanto técnicos como de trabajo en equipo e impacto ético del proyecto	RA1, RA2, RA3
10.b	Presentación 2: - descripción de la organización cliente - problema abordado - trabajo realizado hasta el momento - planificación - reflexión sobre la experiencia	Preparación de la presentación 2	Evaluación sumativa con las ponderaciones correspondientes a la iteración 2 (25% de la calificación final)	RA1, RA2, RA3, RA4

11, 12, 13, 14, 15	- Reunión virtual semanal con el cuerpo docente - Reunión con cliente (es opcional que sea presencial o virtual)	Desarrollo del proyecto Reuniones planificadas y no planificadas con los clientes	En cada reunión semanal se hace una reflexión formativa sobre: - trabajo realizado - planificación de los siguientes pasos - desafíos enfrentados tanto técnicos como de trabajo en equipo e impacto ético del proyecto	RA1, RA2, RA3
--------------------	---	---	--	---------------

Examen	Presentación 3 (examen): - descripción de la organización cliente - problema abordado - trabajo realizado en todo el proyecto - demonstración del producto desarrollado - reflexión sobre la experiencia y los aprendizajes	Preparación de la presentación 3	Evaluación sumativa con las ponderaciones correspondientes a la iteración 3 (50% de la calificación final)	RA1, RA2, RA3, RA4
--------	--	----------------------------------	---	--------------------

D. Estrategias de evaluación:

El curso tiene distintas instancias de evaluación, una para cada iteración:

Actividad evaluada	Tipo de actividad	Evaluador	Ponderación en la nota final			Semana estimada de entrega
			It. 1	It. 2	It 3	
Calidad del software desarrollado	Técnica	Equipo docente	40%	40%	40%	Al final de cada una de las tres iteraciones de desarrollo del proyecto se entrega una calificación a los/las estudiantes, que se calcula como la suma ponderada de cada una de las dimensiones
Gestión de proyecto	Gestión		35%	30%	20%	
Presentación	Comunicación oral		10%	10%	10%	
Valor aportado a la organización	Impacto en el medio	Cliente	0%	10%	25%	
Coevaluación	Trabajo en equipo	Alumnos	15%	10%	5%	

Para aprobar el curso se debe considerar:

- Calificación ponderada final superior a 4. Al final de cada una de las tres iteraciones de desarrollo del proyecto se entrega una calificación a los y las estudiantes que se calcula como la suma ponderada de cada una de las dimensiones.
- Las tres coevaluaciones superiores a 4.
- Asistencia a todas las presentaciones.
- Asistencia al menos al 80% de las reuniones semanales con el cuerpo docente.

Al inicio de cada semestre, el académico o académica informará a los y las estudiantes sobre los tipos de evaluaciones, así como las ponderaciones correspondientes.

E. Recursos bibliográficos:

Bibliografía obligatoria:

- [1] Kent Beck et al. (2001). *Manifesto for Agile Software Development*, <https://agilemanifesto.org/>
- [2] Roger S. Pressman. (2019). *Software engineering: a practitioner's approach*. 8th edition. Mc Graw Hill.
- [3] Ian Sommerville (2017). *Software Engineering*, 10th edition, Pearson.

Bibliografía complementaria:

- [4] D. Strode., T. Dingsøyr, & Y. Lindsjorn. (2022) A teamwork effectiveness model for agile software development. *Empirical Software Engineering*, 27(2), 56.
- [5] M. C. Bastarrica, D. Perovich & M. M. Samary. (2017) What Can Students Get from a Software Engineering Capstone Course? 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET), Buenos Aires, Argentina, 2017, pp. 137-145.
- [6] Y. Lindsjør, D. I.K. Sjøberg, T. Dingsøyr, G. R. Bergersen & T. Dybå. (2016) Teamwork quality and project success in software development: A survey of agile development teams. *Journal of Systems and Software*. Volume 122, Pages 274-286.

F. Datos generales sobre elaboración y vigencia del programa de curso:

Vigencia desde:	Primavera, 2023
Elaborado por:	Cecilia Bastarrica, Daniel Perovich
Validado por:	Académico par validador: Francisco Gutiérrez Validación CTD de Computación, julio 2023.
Revisado por:	Área Gestión Curricular