



Facultad de Ingeniería
Escuela Profesional de Ingeniería Informática

SÍLABO 2026-I

I. DATOS ADMINISTRATIVOS

1. Asignatura	: Taller de Proyectos I
2. Código	: IF0505
3. Naturaleza	: Práctica - Taller
4. Condición	: Obligatorio
5. Requisitos	: Taller de Programación III
6. Nro. Créditos	: 4
7. Nro de horas	: 1 Teóricas/ 6 Prácticas
8. Semestre Académico	: 5
9. Docente	: Javier Añaños, Humberto Linárez, Jesús LLuen, Julio Valverde, Eric Villanueva, Miguel Arrunátegui, Juana Segura, Julián Castro, Yolanda Yopla, ..Miguel Velarde

II. SUMILLA

El curso de Taller de Proyectos I corresponde al quinto ciclo de formación de la Escuela Profesional de Ingeniería Informática. El curso es de naturaleza de taller y tiene como propósito proporcionar al estudiante habilidades que le permitan la integración de las áreas de conocimiento que componen la Ingeniería de software para resolver problemas de Ingeniería informática usando las técnicas, métodos y herramientas apropiadas.

III. COMPETENCIAS GENÉRICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Resolución de Problemas
- Investigación Científica y Tecnológica
- Liderazgo compartido

IV. COMPETENCIAS ESPECÍFICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Solución de Problemas de Ingeniería
- Experimentación y Pruebas
- Trabajo en Equipo
- Práctica Moderna de la Ingeniería

V. DESARROLLA EL COMPONENTE DE: INVESTIGACIÓN () RESPONSABILIDAD SOCIAL ()

VI. LOGRO DE LA ASIGNATURA

Al finalizar el curso, el estudiante podrá desarrollar y mantener sistemas de software confiable y eficiente, que sean económicos, y que satisfagan los requisitos definidos por los clientes. Del mismo modo desarrollarán su versatilidad y adaptabilidad para trabajar e interactuar en los diferentes niveles de un proyecto de ingeniería y con una clara orientación al logro y alcance de objetivos.



VII. PROGRAMACIÓN DE CONTENIDOS

UNIDAD I: Definición del Proyecto		
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante sustenta la organización del equipo y el proyecto a desarrollar. Igualmente sustenta la lista de requisitos que se cubrirán en el proyecto y el plan para la segunda unidad		
Semana	Contenido	
1	Modelar lo esencial del negocio; usar modelo para identificar los CU del sistema; diagramar los CUS.	
2	Especificación detallada de 2 CU de mayor prioridad; lista requisitos funcionales y no funcionales; interpreta Benchmarking.	
3	Plan de alto nivel del proyecto y plan del 2do entregable.	
4	Monitoreo y Retroalimentación. 1er Evaluación del Logro	

UNIDAD II: DEFINICIÓN DE LA ARQUITECTURA		
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante sustenta la arquitectura del sistema propuesto.		
Semana	Contenido	
5	Plantear la arquitectura del sistema (vistas).	
6	Ajustar el alcance; Plan final del proyecto y plan del 3er entregable.	
7	Completar (100%) la especificación de por lo menos los 2 CU de mayor prioridad. Para cada CU se debe hacer: diseño de interfaz de usuario, análisis de los requisitos, modelamiento de BD, diseño, programación, prueba unitaria, integrar al sistema, prueba integral Monitoreo y Retroalimentación. 2da Evaluación del Logro	

UNIDAD III: CONSTRUCCIÓN DEL PRODUCTO		
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante sustenta la construcción, integración y pruebas de por lo menos el 70%.de los casos de uso del producto.		
Semana	Contenido	
9	Correcciones a lo observado en el entregable II.	
10	Continuar con el desarrollo de por lo menos 2 CU adicionales. Plan del 4to entregable.	
11	Completar (100%) la especificación del 70% de los Cus de mayor prioridad. Para cada CU se debe hacer: diseño de interfaz de usuario, análisis de los requisitos, modelamiento de BD, diseño, programación, prueba unitaria, integrar al sistema, prueba integral. Monitoreo y Retroalimentación. 3era Evaluación del Logro	

UNIDAD IV: TRANSICIÓN		
LOGRO DE APRENDIZAJE: Al finalizar la asignatura, el estudiante sustenta el producto terminado, listo para ser entregado a los usuarios.		
Semana	Contenido	
12	Correcciones a lo observado en el entregable III.	
13	Continuar con el desarrollo de por lo menos 2 CU adicionales.	
14	Completar (100%) la especificación del 100% de los Cus de mayor prioridad. Para cada CU se debe hacer: diseño de interfaz de usuario, análisis de los requisitos, modelamiento de BD, diseño, programación, prueba unitaria, integrar al sistema, prueba integral.	



	Monitoreo y Retroalimentación. 4ta Evaluación del Logro
15	EVALUACIÓN FINAL - Sustentación del Proyecto: RÚBRICA

VIII. ESTRATEGIAS DIDÁCTICAS

Las estrategias didácticas a emplear serán el Aula invertida y el Aprendizaje basado en Proyectos.

IX. EVALUACIÓN

UNIDAD	INSTRUMENTOS	PORCENTAJE
I	Rúbrica	5%
II	Rúbrica	10%
III	Rúbrica	20%
IV	Rúbrica	15%
Sustentación del Proyecto	Rúbrica	50%

X. RECURSOS

- Equipos: computadora, laptop, Tablet, celular
- Materiales: apuntes de clase del Docente, separatas de problemas, lecturas, videos.
- IDEs: Visual Studio 2019, Netbeans 9.2, Arduino Studio.

XI. REFERENCIAS

Bibliografía Básica

Sommerville, I. Ingeniería de Software. PEARSON, 9na Edición, 2012.
PRESSMAN, Roger. Ingeniería de Software. McGraw-Hill.; 7ma. Ed., 2010.
RUMBAUGH, J. JACOBSON, I. BOOCH, G. El Lenguaje Unificado de Modelado. Manual de Referencia. Pearson Educación S.A. Madrid, 2000.
RUMBAUGH, J. JACOBSON, I. BOOCH, G. The Unified Software Development Process. Addison Wesley Longman Inc. ISBN 0-201-57169-2, 1999.

Bibliografía complementaria

<http://sce.uhcl.edu/helm/rationalunifiedprocess/> RUP
<http://www.sei.cmu.edu/> Software Engineering Institute Of Carnegie Mellon University
<http://www.isr.uci.edu/> Software Research (ISR).University California, Irvine
<http://www-01.ibm.com/software/rational/> IBM-Rational
<http://www.uml.org/> Lenguaje Uml
<http://www.omg.org/> OMG-Object Management Group
<http://www.bpmn.org> OMG-Business Process Management Initiative
<http://qse.ifs.tuwien.ac.at/index.htm> Quality Software Engineering (QSE) Research at the TU Wien, Institute of Software Technology and Interactive Systems (IFS)
<http://www.ifs.tuwien.ac.at/> Center of Information and Software Engineering
<http://www2.umassd.edu/secenter/saresources.html> Sitios de Recursos de Arquitectura de Software
<http://www.usernomics.com/user-interface-design.html> Principios de diseño de interfaces de usuario
<http://www.ambysoft.com/essays/userInterfaceDesign.html> User Interface Design Tips, Techniques, and Principles
<http://www.loop11.com/> Usabilidad y experiencia de usuarios
<http://www.webstandards.org/learn/tutorials/> Tutoriales de diseño Web
[Software Development in Disruptive Times: Creating a software solution with fast decision capability, agile project management, and extreme low-code technology](#)
[The SPACE of Developer Productivity: There's more to it than you think.](#)
[Best Practice: Application Frameworks: While powerful, frameworks are not for everyone.](#)
[Managing the Hidden Costs of Coordination: Controlling coordination costs when multiple, distributed perspectives are essential](#)
[Revealing the Critical Role of Human Performance in Software: It's time to revise our appreciation of the human side of Internet-facing software systems.](#)