

Programa del curso IC-7841

## **Proyecto de Ingeniería de Software**

Escuela de Computación  
Carrera de Ingeniería en Computación, Plan 411.

## I parte: Aspectos relativos al plan de estudios

### 1 Datos generales

<b>Nombre del curso:</b>	Proyecto de Ingeniería de Software
<b>Código:</b>	IC-7841
<b>Tipo de curso:</b>	Teórico-Práctico
<b>Electivo o no:</b>	No
<b>Nº de créditos:</b>	3
<b>Nº horas de clase por semana:</b>	4
<b>Nº horas extraclase por semana:</b>	5
<b>Ubicación en el plan de estudios:</b>	Curso del VII Semestre del Bachillerato de Ingeniería en Computación
<b>Requisitos:</b>	IC-4302 Bases de Datos II. IC-6831 Aseguramiento de la Calidad del Software
<b>Correquisitos:</b>	AE-4208 Desarrollo de Emprendedores .
<b>El curso es requisito de:</b>	IC-8842 Práctica profesional
<b>Asistencia:</b>	Obligatoria
<b>Suficiencia:</b>	No
<b>Posibilidad de reconocimiento:</b>	Sí
<b>Vigencia del programa:</b>	I Semestre de 2018

## **2 Descripción general**

El curso introduce los conceptos y técnicas necesarios para la creación de productos con tecnologías de tendencia reciente, así como los impactos arquitectónicos que estos puedan tener. El curso presta atención al enfoque de diseño centrado en el usuario, atendiendo especialmente lo concerniente al diseño de la experiencia del usuario. Los estudiantes trabajan en equipos que siguen un enfoque ágil que integra procesos, herramientas y técnicas de análisis, diseño, construcción, verificación y validación de software aprendidos a lo largo de la carrera, para lograr un producto expresado en tecnologías modernas que componen una arquitectura adecuada.

Aplica de manera práctica un proceso de desarrollo de software completo, pasando por todas las etapas – desde el estudio del problema y las necesidades de un cliente real, pasando por el diseño de una solución informática, hasta construir, integrar y probar el producto. Se desarrolla un sistema planteado por los estudiantes del curso, en grupos de 2 o 3 estudiantes.

Este curso resume e integra las mejores prácticas en el desarrollo de sistemas y prepara al estudiante en una de las actividades que puede llegar a desarrollar en el futuro inmediato como profesional en Ingeniería de Software

## **3 Objetivos**

### **Objetivo General**

Desarrollar un sistema que integre conocimientos y mejores prácticas como profesional en Ingeniería de Software.

### **Objetivos Específicos**

1. Gestionar un proyecto de desarrollo de software siguiendo un enfoque ágil en el proceso que enmarque las actividades técnicas.
2. Analizar un problema de desarrollo de software e modelar el dominio en el cual se circunscribe.
3. Describir textualmente los problemas y los requerimientos, tomando en cuenta el contexto, dominio, y limitaciones/restricciones que presenten los usuarios y los interesados.
4. Modelar las clases, las entidades, las relaciones, las operaciones y los comportamientos claves del dominio del problema.
5. Diseñar una arquitectura para solucionar el problema, evaluando alternativas que busquen favorecer los principales atributos de calidad identificados como requerimientos no-funcionales.

6. Aplicar patrones apropiados en los distintos niveles de diseño de la solución.
7. Especificar los componentes de la arquitectura de software propuesta.
8. Construir programas y componentes para satisfacer sus especificaciones.
9. Realizar revisiones sobre documentos técnicos y de código de programación.
10. Planear, diseñar y aplicar pruebas a las unidades, componentes y subsistemas, en todos niveles requeridos, para evaluar su funcionamiento.
11. Integrar los elementos y probar su integración para ensamblar un sistema completo de software que satisfaga los requerimientos según los alcances establecidos.
12. Trabajar en equipos de manera colaborativa, para lograr un producto de alta calidad al final del proyecto.
13. Producir documentación técnica de la especificación, el análisis, el diseño, los componentes, las pruebas y los manuales de usuario.
14. Mantener los documentos técnicos y administrativos que permitan dar visibilidad al trabajo del equipo, controlar sus avances y atender los riesgos.
15. Gestionar rigurosamente las versiones y administrar la configuración de los elementos de software, la documentación y demás recursos que se generen como parte del proyecto.

## **4 Contenidos**

- 1. Enfoques ágiles para el desarrollo de software**
- 2. Definición de los usuarios y dominios para el desarrollo del proyecto**
- 3. Repertorio de tecnologías de software**
  - 3.1. Tecnologías de software para aplicaciones Web
  - 3.2. Tecnologías de software para aplicaciones con dispositivos móviles
  - 3.3. Tecnologías de software para construcción de servicios distribuidos y cómputo en la Nube
- 4. Diseño de la experiencia del usuario**
  - 4.1. Principios de la experiencia del usuario
  - 4.2. Arquitectura de la información
  - 4.3. Paradigmas de interacción

4.4. Diseño de experiencias de usuario

**5. Arquitecturas de aplicaciones de software**

- 5.1. Estilos y patrones arquitectónicos
- 5.2. Arquitecturas multi-capas
- 5.3. Arquitecturas de aplicaciones Web
- 5.4. Servicios y arquitecturas orientadas a servicios
- 5.5. Arquitecturas con dispositivos móviles
- 5.6. Otras consideraciones entre arquitectura y tecnología

**6. Pruebas del software**

**7. Transición e implantación del producto**

- 7.1. Integración y prueba final del producto
- 7.2. Construcción de ayudas en línea
- 7.3. Manuales de usuario
- 7.4. Manuales técnicos
- 7.5. Capacitación del usuario

**8. Gestión calidad del software**

**9. Gestión de la configuración**

## **II parte: Aspectos operativos**

### **5 Metodología de enseñanza y aprendizaje**

La metodología del curso consistirá en la combinación de clases magistrales por parte del profesor, sesiones de seguimiento grupal, exposiciones de avance y de investigaciones, y el trabajo de campo de los estudiantes con sus usuarios.

Las tecnologías y arquitecturas por usar en el curso podrán ser variadas periódicamente, según determine el Consejo de la Escuela de Ingeniería en Computación.

Al inicio del curso se definirán los dominios sobre los cuales los estudiantes realizarán el proyecto

El curso trata de la aplicación práctica de un proceso ágil de desarrollo de software para desarrollar software útil para usuarios reales.

Se brindan los conceptos tecnológicos y metodológicos que complementan el bagaje de conocimientos de los estudiantes, para que puedan construir un producto de software con tecnologías de tendencia reciente.

El proceso de software abarca desde el análisis de un problema hasta la implementación de su solución.

El curso requiere que los estudiantes tengan una gran capacidad de investigación y aprendizaje autónomo.

El profesor les guía en ese proceso, pero es esperable que haya mucha diversidad tecnológica y el profesor no será experto en cada posible tecnología por utilizar.

El proyecto por desarrollar deber tener un alcance definido que no sobrepase las 16 semanas de trabajo  
(Lethbridge & Laganière, 2001) (Augustine, 2005)

## 6 Evaluación

La evaluación está dividida en las diferentes etapas para realizar un producto completo de ingeniería de software tomando con gran interés la interacción con el usuario final así como la planificación y seguimiento de cada estudiante.

	<b>Evaluación</b>	
	<b>Rubro calificación</b>	<b>Puntaje</b>
Informes 35%	Project charter	5%
	Visión	5%
	Prototipo	5%
	Pruebas aceptación del sistema	5%
	Manual técnico	7%
	Calificación final usuario y evidencias seguimiento semanal con el usuario	8%
Proyecto exposiciones 60%	Iteración 1	20%
	Iteración 2	20%
	Iteración 3	20%
Investigación 5%	Paper	5%
	Total:	100%

Calificación iteración 1,2 y 3	
Rubro calificación	Valor
ERS	10
Plan pruebas	5
SAD	10
Ejecución del programa y evidencias del avance personal de cada persona	50
Informe ejecución de pruebas	5
Manual de usuario	10
Carta usuario aceptación	10
Total nota iteración	100

Cronograma de Trabajo		
Semana semestre	Actividad	Fecha aproximada
1, 2, 3	Programa de curso, acuerdo de horario, conformación de equipos de trabajo Primera reunión con cliente	Semana 1 - 3 Del 31 de agosto al 18 de setiembre
3.	Listado de todos los casos de uso del proyecto y propuesta de entregar para cada iteración	Semana 3 Del 14 al 18 de setiembre
4.	Documento de Project charter Documento de Visión del proyecto	Semana 4 21 al 25 de setiembre
5.	Prototipo completo y definición de temas de paper	Semana 5 28 de setiembre al 2 de octubre
6.	Iteración 1- ERS 1 y plan pruebas	Semana 6 5 al 9 de octubre
7.	Iteración 1- SAD	Semana 7 12 al 16 de octubre
8.	Iteración 1- ejecución, manual	Semana 8

		Del 19 al 23 de octubre
9.	Iteración 2- ERS 2 y plan pruebas 2	Semana 9 Del 26 al 30 de octubre
10.	Iteración 2 SAD 2 y entrega de paper	Semana 10 2 al 6 de noviembre
11.	Iteración 2- Ejecución, manual	Semana 11 9 al 13 de noviembre
12.	Iteración 3- ERS 1 y plan pruebas	Semana 12 16 al 20 de noviembre
13.	Iteración 3- SAD	Semana 13 23 al 27 de noviembre
14.	Iteración 3- Ejecución, manual	Semana 14 30 de noviembre al 4 de diciembre
15.	Pruebas sistema	Semana 15 7 al 11 de diciembre
16.	Manual técnico	Semana 16 14 al 18 de diciembre
17.	Calificación usuario y evidencias	Semana 17 18 al 22 enero 2021
<b>Calificación iteración 1,2 y 3</b>		
<b>Rubro calificación</b>		<b>Valor</b>
ERS*		10
Plan pruebas		5
SAD*		10
Ejecución del programa y evidencias del avance personal de cada persona		50
Informe ejecución de pruebas		5
Manual de usuario		10



Carta usuario aceptación	10
Total nota iteración	100
<p>* Estos documentos se entregan según metodología de desarrollo seleccionada, si es ágil se cambian los entregables por alguno que se adapte a la metodología.</p>	

### Disposiciones generales

El proyecto por desarrollar deber tener un alcance definido que no sobrepase las 16 semanas de trabajo, con una dedicación de 9 horas de trabajo por cada miembro de equipo.

1. El curso se aprueba con nota de 67
2. NO hay examen de reposición
3. La entrega de cada documento del curso, se debe colocar en el sitio de TEC digital del curso o carpeta de drive, a más tardar el día asignado antes de la medianoche.
4. PAPER: Cada estudiante deberá realizar un paper con los temas relacionados con los temas del proyecto y del curso, los cuales se asignarán en la semana 2 del período de verano y se entregará a más tardar en semana 5 del mismo.
5. PROYECTO: El tamaño se mide por casos de uso. Cada alumno debe desarrollar al menos 5 casos de uso. Uno de los cuales uno debe ser para dispositivo móvil. Para esta medición no se toman en cuenta los casos de uso de gestión de usuarios, seguridad ni login, pero si deben implementarse. Se desarrollar en forma iterativa.
6. PROJECT CHARTER y VISION: se debe entregar al profesor en digital
7. PROTOTIPO: debe ser realizado en algún software libre y multiplataforma, debe ser un prototipo navegable, no solo mockups.

En el desarrollo del prototipo se debe abarcar todos los casos de uso del proyecto, adjuntar carta aprobación usuario. Esta carta valdrá 30 puntos de la calificación del prototipo.

8. PARA CADA ITERACION: Los documentos ERS, SAD, plan e informe pruebas deben ser diferentes pues varían los casos de uso de cada iteración.

9. MINUTAS Y/O EVIDENCIAS DE COMUNICACIÓN CON EL USUARIO: deben tener el formato dado al inicio del curso para una minuta o el mecanismo formal de interacción con el usuario, para las entregas si debe venir con firma y sello del usuario. Se debe presentar solamente una minuta o evidencia por semana del semestre.

A las reuniones con el usuario deben asistir todos los integrantes del grupo, por lo que las minutas deben venir firmadas por el usuario y por cada estudiante. Estas evidencias deben corresponder a reuniones con el usuario, en los que se traten temas del proyecto.

Minutas para firmar documentos no se toman en cuenta dentro del mínimo total por entregar.

Se debe generar un pdf evidencia semanal de contacto con el usuario se deben subir al TEC Digital cada semana o carpeta de drive, si se utiliza otro medio debe indicarse.

10. EVIDENCIAS DE AVANCE SEMANAL INDIVIDUAL: la bitácora individual almacenarse en GitHub, debe haber evidencia del avance de cada miembro según tareas asignadas, este rubro es de suma importancia para la calificación de cada iteración.

11. CALIFICACION FINAL USUARIO: es indispensable para aprobar el proyecto. Debe ser un email o un documento impreso de parte del usuario, con una calificación del 1 al 10 y con observaciones para oportunidades de mejora que justifiquen la nota asignada

12. CALIFICACION GRUPAL: es una calificación de parte de cada alumno hacia sus compañeros de grupo de trabajo. El promedio de esta calificación se multiplicará por la calificación obtenida en cada iteración grupal, para definir la nota de la iteración.

13. ENTREGA DE CALIFICACION GRUPAL: Debe ser enviada por correo electrónico al profesor, el mismo día que se entrega el manual técnico

14. CALIFICACION ITERACION: La calificación de las dos iteraciones grupales será el producto de la calificación de los compañeros de grupo multiplicado por la nota obtenida por el grupo en los tópicos de la iteración

15. Correo electrónico: todo correo electrónico del curso debe llevar al inicio del "asunto" lo siguiente:

IC-7841-Proy-", nombre del documento o punto de control, y nombre del estudiante. Debe ir dirigido a la cuenta de la profesora del curso: mariaestrada.s@gmail.com

16. SOBRE DOCUMENTOS IMPRESOS: todo entregable será alojado en la fecha de entrega en el TECdigital, carpeta de drive o bien en el repositorio de github, al finalizar el curso según requerimiento del usuario final se imprimirán los documentos, de lo contrario se entregará todo el proyecto completo y documentos en algún dispositivo portable para entregarse al usuario.

17. LUGAR REVISION: se realizará de manera virtual según se acuerde previamente.

## **7 Bibliografía**

Lethbridge, Timothy; Laganière, Robert. Object-Oriented Software Engineering. McGraw Hill. 2001. ISBN 0-07-710908-2.

Augustine, S. (2005). Managing Agile Projects. Prentice-Hall.

Rosenberg, Doug, Stephens, Matt (2007). Use Case Driven Object Modeling with UML: Theory and Practice. Springer-Verlag, New York, USA.

## **8 Profesora**

Ing María Estrada Sánchez MSc.

Horario: Se atenderá según horario que se acuerde con cada equipo.

Horario de consulta: lunes 10 am-12 md o en otro horario según se acuerde con cada equipo (coordinar previamente)

Otros medios de consulta:

Telegram: <https://telegram.me/MariaEstrada>

Skype: mariaestrada.s

Correo electrónico mariaestrada.s@gmail.com

Teléfono: +506 8323-2214