

# UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

## COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL

### PROGRAMA DE UNIDAD DE APRENDIZAJE

#### I. DATOS DE IDENTIFICACIÓN

- 1. Unidad Académica:** Facultad de Ingeniería, Arquitectura y Diseño, Ensenada; Facultad Ciencias Químicas e Ingeniería, Tijuana; y Facultad de Ciencias de la Ingeniería y Tecnología, Valle de las Palmas
- 2. Programa Educativo:** Ingeniero en Software y Tecnologías Emergentes
- 3. Plan de Estudios:** 2022-1
- 4. Nombre de la Unidad de Aprendizaje:** Administración de Proyectos de Software
- 5. Clave:** 40020
- 6. HC: 02 HT: 00 HL: 02 HPC: 00 HCL: 00 HE: 02 CR: 06**
- 7. Etapa de Formación a la que Pertenece:** Terminal
- 8. Carácter de la Unidad de Aprendizaje:** Obligatoria
- 9. Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



#### Equipo de diseño de PUA

Luis Ángel Monge De la Cruz  
María Angélica Astorga Vargas  
Manuel Castañón Puga  
J. Reyes Juárez

#### Vo.Bo. de subdirector(es) de Unidad(es) Académica(s)

Humberto Cervantes De Ávila  
Daniela Mercedes Martínez Platas  
Noemí Hernández Hernández

**Fecha:** 20 de febrero de 2021

## **II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE**

La unidad de aprendizaje Administración de Proyectos de Software tiene como finalidad que el alumno conozca y sea capaz de poner en práctica los conceptos asociados a la administración de proyectos de software; así como desarrollar habilidades de liderazgo, planeación, gestión de recursos, y monitoreo de actividades, de manera analítica, con visión de actualización, empatía, actitud crítica y respetuosa.

Esta unidad de aprendizaje es de carácter obligatorio, se encuentra dentro de la etapa terminal y forma parte del área de conocimiento de aspectos organizacionales y TI de la licenciatura Ingeniero en Software y Tecnologías.

## **III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE**

Administrar el ciclo de vida de un proyecto de desarrollo de software, a través de herramientas y apego a estándares internacionales que guíen de manera sistemática las actividades de planificación, manejo de riesgos, control de calidad y control de versiones para asegurar que se cumpla con el tiempo y costo estimados de acuerdo a los requerimientos inicialmente establecidos, con actitud proactiva, creativa y de liderazgo.

## **IV. EVIDENCIA(S) DE APRENDIZAJE**

Elabora plan de gestión y ejecución de un proyecto de Software que incluya la definición del proyecto, análisis de factibilidad, plan de trabajo, estimación de tiempo, esfuerzo, costos y recursos, integración de equipo de trabajo, plan de administración de riesgos y plan de liberación.

**V. DESARROLLO POR UNIDADES**  
**UNIDAD I. Introducción a la administración de proyectos de software**

**Competencia:**

Analizar los conceptos básicos de la administración de proyectos de software, a través del análisis de su propósito e importancia, con la finalidad de reconocer su aplicación, mostrando actitud analítica y pensamiento crítico.

**Contenido:**

**Duración:** 4 horas

- 1.1 Naturaleza de los proyectos de software
- 1.2 Modelos de desarrollo de software: el papel de la administración
  - 1.2.1 Los modelos tradicionales
  - 1.2.2 Las metodologías ágiles
  - 1.2.3 Los modelos mixtos
- 1.3 Los roles y tareas en los procesos de desarrollo de software
  - 1.3.1 Roles administrativos
  - 1.3.2 Roles técnicos
- 1.4 Las fases de la administración de proyectos de software
  - 1.4.1 Las fases genéricas de la administración
  - 1.4.2 Las fases de la administración de un proyecto de software
- 1.5 Las competencias del rol administrador de proyectos de software
  - 1.5.1 La personalidad
  - 1.5.2 Las capacidades de gestión
  - 1.5.3 Las habilidades de comunicación
  - 1.5.4 La actitud de liderazgo

## UNIDAD II. Planeación de un proyecto de software

### Competencia:

Determinar la estrategia de ejecución de un proyecto de software, a través del análisis de su alcance y características empleando técnicas y herramientas pertinentes, para su correcta administración, con actitud crítica, analítica y propositiva.

### Contenido:

**Duración:** 6 horas

- 2.1 Especificación del proyecto
  - 2.1.1 Fuentes de ideas/necesidades para generar un proyecto
  - 2.1.2 Determinación de la naturaleza de un proyecto
  - 2.1.3 Criterios para la selección de un proyecto
  - 2.1.4 Técnicas de especificación de proyectos
- 2.2 Características del proyecto
  - 2.2.1 Los requerimientos del sistema (“product backlog”)
  - 2.2.2 Alcance y objetivos del proyecto
  - 2.2.3 Definición infraestructura
  - 2.2.4 Definición de características y actividades
  - 2.3.5 La determinación de recursos humanos (equipo de trabajo)
- 2.3 Viabilidad y factibilidad del proyecto
  - 2.3.1 Viabilidad técnica, económica, operacional
  - 2.3.2 Factibilidad técnica económica, operacional
- 2.4 Plan de ejecución
  - 2.4.1 Calendarización del Proyecto
  - 2.4.2 Disposición de recursos
  - 2.4.3 Estrategias de comunicación
- 2.5 Herramientas para la planeación de proyectos

## UNIDAD III. La estimación de un proyecto de software

### Competencia:

Estimar los recursos requeridos para el desarrollo de un proyecto de software, a través del análisis de sus necesidades empleando técnicas y herramientas pertinentes y con apego a estándares, para determinar el costo del proyecto, con actitud crítica, analítica y honesta.

### Contenido:

**Duración:** 8 horas

- 3.1 Estimación de tiempo
  - 3.1.1 El tiempo como factor base del desarrollo de un proyecto
  - 3.1.2 Técnicas de estimación de la duración de un proyecto
- 3.2 Estimación de esfuerzo
  - 3.2.1 Esfuerzo vs complejidad
  - 3.2.2 Técnicas de estimación de esfuerzo
- 3.3 Estimación de costos
  - 3.3.1 Tipos de costos
  - 3.3.2 Técnicas y modelos de estimación de costos
- 3.4 Manejo de riesgos e incertidumbres
  - 3.4.1 Los tipos de incertidumbres en los proyectos de software
  - 3.4.2 Identificación de riesgos
  - 3.4.3 Manejo de riesgos
  - 3.4.4 Planes de contingencia
- 3.5 Herramientas y técnicas para la estimación de proyectos
  - 3.5.1 Uso de datos históricos
  - 3.5.2 Técnicas de estimación basadas en consenso
  - 3.5.1 Herramientas software para estimación

## UNIDAD IV. Ejecución de un proyecto de software

### Competencia:

Administrar los recursos asignados a un proyecto de software, a través del análisis del seguimiento de las actividades del proyecto empleando técnicas y herramientas pertinentes, para asegurar el cumplimiento de sus metas, con actitud crítica, analítica y empática.

### Contenido:

**Duración:** 8 horas

- 4.1 Gestión de requisitos
  - 4.1.1 Administración del “producto backlog”,
  - 4.1.2 Manejo de prioridades, dependencias
  - 4.1.3 Control de cambios
- 4.2. Supervisión de los equipos de trabajo
  - 4.2.1 Autogestión del equipo (seguimiento del progreso, asignación de carga de trabajo)
  - 4.2.2 Respuesta a problemas emergentes
  - 4.2.3 Dinámica y psicología de grupos
  - 4.2.4 Métricas de desempeño del equipo de trabajo
- 4.3. Monitoreo de progreso del proyecto
  - 4.3.1 Valor ganado, velocidad, gráficos de quemado
  - 4.3.2 Seguimiento de defectos
  - 4.3.3 Gestión de la deuda técnica
  - 4.3.4 Métricas de rendimiento del proyecto
- 4.4 Administración del tiempo
  - 4.4.1 Índice de cumplimiento del cronograma
  - 4.4.2 Administración de ajustes al cronograma del proyecto
- 4.5 Creación de reportes de “status” del proyecto
- 4.6 Herramientas software para la administración de la ejecución de proyectos

## UNIDAD V. Monitoreo y control de un proyecto de software

### Competencia:

Analizar los resultados del desarrollo de un proyecto de software, por medio del monitoreo y revisión de las actividades realizadas empleando técnicas y herramientas pertinentes y con apego a estándares, para asegurar la liberación de un producto de calidad, mostrando actitud crítica, honesta y ética profesional.

### Contenido:

**Duración:** 6 horas

- 5.1 Plan de liberación de proyecto
- 5.2 Administración de la configuración del software
  - 5.2.1 Técnicas de configuración de software
  - 5.2.2 Control de versiones
  - 5.2.3 Herramientas de administración de la configuración
- 5.3 Monitoreo y Control
  - 5.3.1 Monitoreo y control de riesgos
  - 5.3.2 Monitoreo y control de costo
  - 5.3.3 Monitoreo y control de esfuerzo
- 5.4 Administración y seguimiento de pruebas
  - 5.4.1 Tipos de pruebas
  - 5.4.2 Pruebas de integración
  - 5.4.3 Pruebas de integración

## VI. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

No.	Nombre de la Práctica	Procedimiento	Recursos de Apoyo	Duración
<b>UNIDAD I</b>				
1	<b>Definición de roles y actividades</b>	<ol style="list-style-type: none"> <li>1. Identifica y analiza un problema real que pueda resolverse por medio de un sistema de software.</li> <li>2. Establece/considera los requerimientos técnicos del sistema a implementar y del proyecto.</li> <li>3. Identifica las personas que participarán en el proyecto, los roles y las actividades que desarrollarán, así como las habilidades que requieren tener, con base a los requerimientos técnicos del sistema a implementar.</li> <li>4. Realiza un documento con un reporte con la descripción del proyecto y los participantes y sus actividades.</li> <li>5. Entrega el documento al docente para su evaluación.</li> </ol>	<ul style="list-style-type: none"> <li>● Papel o cuaderno</li> <li>● Lápiz o pluma</li> <li>● Computadora</li> <li>● Software de procesamiento de texto.</li> </ul>	2 horas
<b>UNIDAD II</b>				
2	<b>Planeación de un proyecto de software</b>	<ol style="list-style-type: none"> <li>1. Revisa las personas que participarán en el proyecto conforme a las actividades.</li> <li>2. Analiza los requerimientos y el alcance del proyecto.</li> <li>3. Analiza la viabilidad y factibilidad del proyecto.</li> </ol>	<ul style="list-style-type: none"> <li>● Papel o cuaderno</li> <li>● Lápiz o pluma</li> <li>● Computadora</li> <li>● Software de procesamiento de texto.</li> <li>● Herramientas de planeación del proyecto</li> </ul>	6 horas

		<ol style="list-style-type: none"> <li>4. Establece el plan de ejecución del proyecto, considerando actividades, roles, tiempo y recursos.</li> <li>5. Realiza un documento con un reporte con la descripción de la planeación del proyecto.</li> <li>6. Entrega el documento al docente para su evaluación.</li> </ol>		
<b>UNIDAD III</b>				
3	<b>Estimación de un proyecto de software</b>	<ol style="list-style-type: none"> <li>1. Revisa la planeación del proyecto.</li> <li>2. Identifica los recursos necesarios para el correcto desarrollo del proyecto, empleando las técnicas y herramientas pertinentes y con apego a estándares.</li> <li>3. Realiza un documento con un reporte con la descripción de la planeación del proyecto.</li> <li>4. Entrega el documento al docente para su evaluación.</li> </ol>	<ul style="list-style-type: none"> <li>● Papel o cuaderno</li> <li>● Lápiz o pluma</li> <li>● Computadora</li> <li>● Software de procesamiento de texto.</li> <li>● Herramientas de planeación del proyecto</li> </ul>	8 horas
<b>UNIDAD IV</b>				
4	<b>Ejecución de un proyecto de software</b>	<ol style="list-style-type: none"> <li>1. Revisa las actividades, recursos y compromisos del proyecto.</li> <li>2. Compara los resultados del desarrollo del proyecto con respecto a lo establecido en el plan de ejecución del proyecto.</li> </ol>	<ul style="list-style-type: none"> <li>● Papel o cuaderno</li> <li>● Lápiz o pluma</li> <li>● Computadora</li> <li>● Software de procesamiento de texto.</li> <li>● Herramientas de planeación del proyecto</li> </ul>	8 horas

		<ol style="list-style-type: none"> <li>3. Realiza un documento con un reporte con el análisis del avance del proyecto.</li> <li>4. Entrega el documento al docente para su evaluación.</li> </ol>		
<b>UNIDAD V</b>				
5	<b>Control de un proyecto de software</b>	<ol style="list-style-type: none"> <li>1. Revisa el avance del proyecto, empleando las técnicas y herramientas pertinentes para el seguimiento y control.</li> <li>2. Determina la calidad de los entregables del proyecto empleando las técnicas y herramientas pertinentes, así como la utilización de los recursos asignados.</li> <li>3. Realiza un documento con un reporte con el análisis del cumplimiento del proyecto.</li> <li>4. Entrega el documento al docente para su evaluación.</li> </ol>	<ul style="list-style-type: none"> <li>● Papel o cuaderno</li> <li>● Lápiz o pluma</li> <li>● Computadora</li> <li>● Software de procesamiento de texto.</li> <li>● Herramientas de planeación del proyecto</li> </ul>	8 horas

## VII. MÉTODO DE TRABAJO

**Encuadre:** El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

### **Estrategia de enseñanza (docente):**

- Técnica expositiva
- Fomenta el trabajo colaborativo
- Plantea escenario de administración de proyecto.
- Guía en diseño, desarrollo y documentación del proyecto
- Guías prácticas de taller y laboratorio
- Realiza evaluaciones
- Considera la retroalimentación

### **Estrategia de aprendizaje (alumno):**

- Búsqueda de información sobre la administración de proyectos de software
- Búsqueda de escenarios y sistemas para planteamiento y administración de proyectos
- Analiza los criterios y metodologías para la administración de proyectos
- Redacta protocolos de planeación y seguimiento de proyectos
- Realiza prácticas de laboratorio
- Trabaja de manera colaborativa
- Diseña instrumentos para la recolección de datos para el seguimiento y control de proyectos
- Redacta reportes técnicos sobre la administración de proyectos de software

## VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

### **Criterios de acreditación**

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

### **Criterios de evaluación**

- Evaluaciones parciales..... 40%
- Prácticas de taller y laboratorio..... 30%
- Reporte de plan de gestión de proyecto..... 30%
- Total.....100%**

## IX. REFERENCIAS

### Básicas

- Golden, W. (2019). *Project Management: An Essential Guide for Beginners Who Want to Understand Agile, Scrum, Lean Six Sigma, Kanban and Kaizen When Applied to Managing Projects*. Estados Unidos: Bravex Publications
- Kloppenborg, T, Anantatmula, V. S., Wells, K. (2018). *Contemporary Project Management*; (4a ed.). Estados Unidos: Cengage Learning .
- Project Management Institute. (2017). *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*—(6a ed.). Estados Unidos: Project Management Institute.
- Project Management Institute. (2017). *Agile Practice Guide, (new edition)*. Estados Unidos: Project Management Institute.
- Gido, J., Clements, J., Baker, R. (2017). *Successful Project Management*. (7a ed.). Estados Unidos: Cengage Learning
- Schwalbe, K. (2017). *An Introduction to Project Management*, (6a ed.). Estados Unidos: CreateSpace Independent Publishing Platform

### Complementarias

- Evans, J. R., Lindsay, W. M. (2016). *Managing for Quality and Performance Excellence*. (10a ed.). Estados Unidos: Cengage Learning
- Kerzner, H. (2005). *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, (9a ed.). Estados Unidos: Wiley [clásica]

## **X. PERFIL DEL DOCENTE**

El docente que imparta esta asignatura debe poseer nivel licenciatura en ingeniería afín a la ciencia de la computación, preferiblemente con posgrado (maestría y doctorado) en un área a fin. Experiencia docente y experiencia práctica en el campo disciplinar de ingeniería de software y administración de proyectos de software. Con capacidad de abstracción para análisis y diseño de sistemas software, habilidades de dirección de proyectos, comunicación, motivación, emprendimiento e innovación.