Aprobado por resolución Nº113 del CFI de fecha 04.07.2017

FACULTAD DE INGENIERIA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Programa de

Taller de Planificación y Seguimiento de Proyectos de Software

1. NOMBRE DE LA UNIDAD CURRICULAR

Taller de Planificación y Seguimiento de Proyectos de Software

2. CRÉDITOS

4 créditos.

3. OBJETIVOS DE LA UNIDAD CURRICULAR

Que el estudiante aprenda los conceptos y domine las técnicas básicas necesarias para llevar adelante la planificación y el seguimiento de un proyecto de software.

4. METODOLOGÍA DE ENSEÑANZA

El curso consiste en clases teórico-prácticas, lecturas domiciliares obligatorias, controles de lectura en línea, ejercicios prácticos domiciliarios semanales, y presentaciones de los estudiantes.

Las clases teórico-prácticas tendrán una duración de 2 horas, y habrá dos clases semanales durante siete semanas y media, totalizando 15 clases y 30 h presenciales distribuidas de la siguiente manera:

Horas de teórico = 23
Horas de práctico = 7

Además, se requiere la participación de los estudiantes en:

- Presentaciones.
- Entrega semanal de ejercicios prácticos domiciliarios.

Se estima que cada de estudiante deberá dedicar un total de 21 horas a ejercicios prácticos domiciliarios, 6 horas a la lectura de los materiales indicados y 3 horas de investigación y preparación de la presentación, totalizando 30 horas de dedicación no presencial estimada.

5. TEMARIO

1. **Introducción**
   - Visión
   - Contexto del desarrollo de software, su evolución, tendencias, distintos tipos de proyectos
   - Tipos de proceso y planificación del proyecto
   - El proceso en cascada y sus problemas
   - El costo del cambio
   - Los espacios del problema y de la solución
   - El desarrollo en fases
   - El desarrollo guiado por planes y el desarrollo adaptativo
2. **Plan para la dirección del proyecto**
   - Contenidos. Estándares.
   - Técnicas para el armado del plan
   - Escenarios para desarrollar un plan
3. **Gestión del alcance**
   - Alcance del producto y alcance del proyecto
   - ADV
   - WBS
   - Determinación del alcance (otros parámetros).
4. **Plan de la iteración**
   - Concepto de iteraciones, liberaciones
   - Coordinación entre los distintos planes
5. **El proceso de planificación**
   - Priorización de requisitos
   - Riesgos en proyectos de software y su relación con la duración y tamaño de los proyectos
   - Participación de las áreas interesadas en la planificación y el seguimiento
6. **Secuenciar las actividades**
   - Grafo de actividades
   - Método de diagramación por precedencias (PDM)
7. **Desarrollar el cronograma**
   - Técnicas:
     - Método del Camino Crítico
     - Método de la Cadena Crítica
     - PERT
     - Diagrama de Gantt
   - Técnicas de optimización de recursos:
     - Perfil de uso de recursos
     - Nivelación de recursos
     - Equilibrio de recursos
   - Técnicas de modelado
     - Análisis «¿Qué pasa si…?»
     - Simulación.
     - El modelo de Abdel-Hamid y variaciones.
   - Características del software que inciden en la planificación y seguimiento.

- Técnicas de compresión del cronograma
- *Time-boxing*

8. Reflexiones

- Cuándo planificar
- Planes por períodos o por producto
- Requisitos de un buen plan
- Frecuencia de planificación
- Mantenimiento del plan

9. Controlar el cronograma. Registro y control de avance

- Técnicas de registro y control de avance del proyecto:
  - Diagrama de Gantt
  - Diagrama de evolución de gastos
  - Enfoque del valor ganado.
- Registro de esfuerzo
- Análisis del esfuerzo:
  - Real vs. planificados. Cuánto le erré
  - Por qué le erré: mala estimación (más o menos) / no se cumplió la actividad
- Técnicas de medición de avance de actividades
- Distinción entre avance real y percibido.
- Verificación y validación y su relación con el seguimiento

10. Planificación y seguimiento del avance en procesos ágiles

11. Herramientas

6. BIBLIOGRAFÍA

Tema	Básica	Complementaria
1. Introducción • Tipos de proceso y planificación del proyecto • Visión	(1)	(13)
2. Plan para la dirección del proyecto • Contenidos • Técnicas para el armado del plan • Escenarios para desarrollar un plan	(1)	(7)
3. Gestión del alcance • Alcance del producto y alcance del proyecto • ADV • WBS • Determinación del alcance (otros parámetros)	(1) y (2)	(7) y (8)
4. Plan de la iteración • Concepto de iteraciones, liberaciones • Coordinación entre los distintos planes	(1)	
5. El proceso de planificación	(1)	(7)
6. Secuenciar las actividades	(1)	

• Grafo de actividades • Método de diagramación por precedencias (PDM)		
7. Desarrollar el cronograma • Técnicas: o Método del Camino Crítico (CPM) o Método de la Cadena Crítica (CCM) o PERT o Diagrama de Gantt • Técnicas de optimización de recursos: o Perfil de uso de recursos o Nivelación de recursos o Equilibrio de recursos • Técnicas de modelado o Análisis «¿Qué pasa si…?» o Simulación • Técnicas de compresión del cronograma • *Time-boxing*	(1) y (5)	(7) y (11)
8. Reflexiones • Cuándo planificar • Planes por períodos o por producto • Requisitos de un buen plan • Frecuencia de planificación • Mantenimiento del plan	(1)	(6), (9) y (10)
9. Controlar el cronograma. Registro y control de avance. • Técnicas de registro y control de avance del proyecto: o Diagrama de Gantt o Diagrama de evolución de gastos o Enfoque del valor ganado. • Registro de esfuerzo • Análisis del esfuerzo • Técnicas de medición de avance de actividades	(1) y (3)	
10. Planificación y seguimiento del avance en procesos ágiles	(4)	(7), (12) y (13)
11. Herramientas	(1)	

6.1 Básica

1. Project Management Institute (PMI) (2013). *Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK)*, 5ª edición. Pensilvania: Project Management Institute, Inc.
2. PMI (2006). *Practice Standard for Work Breakdown Structures*, 2ª edición. Pensilvania: Project Management Institute, Inc.
3. PMI (2005). *Practice Standard for Earned Value Management*. Pensilvania: Project Management Institute, Inc.
4. Cohn, M. (2006). *Agile Estimating and Planning*. Upper Saddle River, New Jersey: Pearson Education, Inc.

5. Miranda, E. (2011). Time boxing planning: buffered Moscow Rules. *ACM SIGSOFT Software Engineering Notes* 36(6):1-5.

6.2 Complementaria

6. Humphrey, W. S. y Thomas, W. R. (2010). *Reflections on Management. How to manage your software projects, your teams, your boss and yourself*. Boston: Pearson Educations, Inc.
7. Fairley, R. E. (2000). *Managing and leading software projects*. IEEE Computer Society. Hoboken, New Jersey: John Wiley & Sons, Inc.
8. Hall Thayer, R., y Dorfman, M. (eds.) (2013). *Software Engineering Essentials, vol. II: The supporting processes*, 4º edición. California: Software Management Training.
9. Riaz Ahamed, S. S. (2010). «Project Planning: An Analysis». *International Journal of Engineering Science and Technology* 2(1).
10. Zwikael, O. y Globerson, S. (2004). «Evaluating the Quality of Project Planning: A Model and Field Results». *International Journal of Production Research* 42(8) (p. 1545-1556).
11. Goldratt, E. M. (2001 [1997]). *Cadena crítica. Una novela empresarial sobre la gestión de proyectos*. Ediciones Díaz de Santos, S. A.
12. Schwaber, K., y Sutherland, J. (2017). *La guía de Scrum: las reglas del juego*.
13. Sommerville, I. (2016), *Software Engineering*, 10ª edición. Harlow: Pearson Education.

7. CONOCIMIENTOS PREVIOS EXIGIDOS Y RECOMENDADOS

7.1 Conocimientos previos exigidos:

Conocimientos básicos de Ingeniería de Software.

7.2 Conocimientos previos recomendados: -----

ANEXO A
Para todas las carreras

Esta primera parte del anexo incluye aspectos complementarios que son generales de la unidad curricular.

A1) INSTITUTO

Instituto de Computación

A2) CRONOGRAMA TENTATIVO

El curso se desarrollará durante ocho semanas (15 clases).

Semana 1	Introducción (2 h de clase).
Semana 2	Plan para la dirección del proyecto (4 hs de clase).
Semana 3	Gestión del alcance. ADV (2 h de clase). WBS (2 h de clase).
Semana 4	Plan de la iteración (2 h de clase). El proceso de planificación (2 h de clase).
Semana 5	Secuenciar las actividades (2 h de clase). Desarrollar el cronograma. CPM, CCM y PERT (2 h de clase).
Semana 6	Desarrollar el cronograma Gantt. Técnicas de modelado, optimización y compresión del cronograma (2 h de clase). Time-boxing (1 h de clase). Reflexiones (1 h de clase).
Semana 7	Controlar el cronograma (2 h de clase). Planificación y seguimiento en procesos ágiles (2 h de clase).
Semana 8	Planificación y seguimiento en procesos ágiles (2 h de clase). Herramientas (2 h de clase).

A3) MODALIDAD DEL CURSO Y PROCEDIMIENTO DE EVALUACIÓN

Metodología de evaluación:

- controles de lectura en línea a través de EVA
- informes de lectura a partir de artículos
- ejercicios domiciliarios
- presentaciones en clase

La aprobación requiere:

- 75 % de la asistencia a las clases teórico-prácticas.
- entrega de todos los ejercicios domiciliarios y un puntaje mínimo de aprobación en cada uno del 60 %.
- Un promedio de 70 % en los controles e informes de lectura.
- un mínimo aceptable en las presentaciones.

La nota final estará conformada en un 60 % por el promedio de los ejercicios domiciliarios., en un 25 % por los controles e informes de lectura y un 15 % por las presentaciones.

A4) CALIDAD DE LIBRE

Los estudiantes no podrán acceder a la calidad de libre.

A5) CUPOS DE LA UNIDAD CURRICULAR

Cupos mínimos: no tiene
Cupos máximos: 30 estudiantes

Aprobado por resolución Nº113 del CFI de fecha 04.07.2017

ANEXO B para las carreras de Ingeniería en Computación (plan 97) y Licenciatura en Computación.

B1) ÁREA DE FORMACIÓN

Ingeniería de Software

B2) UNIDADES CURRICULARES PREVIAS

Para el curso: Examen aprobado de Introducción a la Ingeniería de Software

Para el examen: no aplica