



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

GUÍA DOCENTE DE SEMINARIO-TALLER DE SOFTWARE

La presente guía docente corresponde a la asignatura Seminario-Taller de Software (STI-S) del Grado en Ingeniería Informática, aprobada para el curso lectivo 2011-2012 en Junta de Centro y publicada en su versión definitiva en la página web de la Escuela Politécnica Superior. La guía docente de STI-S aprobada y publicada antes del periodo de matrícula tiene el carácter de contrato con el estudiante.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

ASIGNATURA

SEMINARIO-TALLER DE SOFTWARE (STI-S)

1.1. Código

17818 de la titulación de Grado en Ingeniería Informática

1.2. Materia

Seminario-Taller de Informática

1.3. Tipo

Formación obligatoria

1.4. Nivel

Grado

1.5. Curso

1º

1.6. Semestre

1º

1.7. Número de créditos

3 créditos ECTS (1,25 presenciales + 1,75 no presenciales)

1.8. Requisitos previos

CONOCIMIENTOS Y HABILIDADES PREVIOS QUE PERMITEN ENCARAR EL APRENDIZAJE DE LA ASIGNATURA

Es requisito para la correcta adquisición de los contenidos, habilidades y destrezas de la asignatura tener buena capacidad de: lectura comprensiva de textos en español e inglés, búsqueda e identificación de bibliografía y redacción de memorias en la lengua española. Asimismo, se requiere una buena disposición tanto para el aprendizaje y desarrollo de la disciplina personal de estudio como para el trabajo en equipo.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

RECOMENDACIONES

Para superar con éxito la asignatura Seminario-Taller de Software es muy recomendable estar cursando la asignatura Programación I, donde se adquiere un enfoque algorítmico para la resolución de problemas y conocimientos de C.

Además, se considera necesaria la asistencia continua a las sesiones de prácticas, de grupo, de resolución de problemas, de manejo de herramientas y de construcción de sistemas de software.

Por último, es recomendable disponer de un nivel de inglés que permita al estudiante leer bibliografía de consulta.

CORRELATIVAS

Las asignaturas Seminario-Taller de Software y Seminario-Taller de Hardware conforman la materia Seminario-Taller de Informática de la titulación de Grado en Ingeniería Informática.

La asignatura Seminario-Taller de Software está transversalmente relacionada con la asignatura Programación I, porque complementa los conocimientos, habilidades y actitudes adquiridos en dicha asignatura. STI-S sirve de base para los conocimientos, habilidades y actitudes que se adquirirán en las asignaturas de Programación II y Proyecto de Programación del segundo cuatrimestre. Este conjunto de asignaturas confluyen en resultados de aprendizaje, competencias y contenidos.

1.9. Requisitos mínimos de asistencia a las sesiones presenciales

Se plantean dos métodos de evaluación: evaluación continua (con asistencia obligatoria a clase) y evaluación no continua (sin asistencia obligatoria a clase). Los estudiantes deberán optar por uno u otro método a principio del curso (dos primeras semanas) y cumplir con los distintos requisitos de evaluación que conlleva cada uno de los modelos, publicados en la presente guía docente (véase apartado 4).

EVALUACIÓN CONTINUA CON ASISTENCIA OBLIGATORIA A CLASE

La asistencia es obligatoria al menos en un 85%.

EVALUACIÓN NO CONTINUA SIN ASISTENCIA OBLIGATORIA A CLASE

La asistencia es obligatoria al menos en un 30%. Se debe estar presente en las sesiones de entrega de cada práctica para su evaluación oral parcial correspondiente sobre la práctica realizada.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

1.10. Datos del equipo docente

Nota: se debe añadir @uam.es a todas las direcciones de correo electrónico.

Dra. Silvia Teresita Acuña

Departamento de Ingeniería Informática
Escuela Politécnica Superior
Despacho - Módulo: B-331 Edificio B - 3ª Planta
Teléfono: +34 91 497 2275
Correo electrónico: silvia.acunna
Página web: <http://arantxa.ii.uam.es/~sacuna/is1/>
Horario de atención al alumnado: Petición de cita previa por correo electrónico.

Dr. Carlos Santa Cruz Fernández (Coordinador)

Departamento de Ingeniería Informática
Escuela Politécnica Superior
Despacho - Módulo: B-343 Edificio B - 3ª Planta
Teléfono: +34 91 497 2337
Correo electrónico: carlos.santacruz
Página web: <http://arantxa.ii.uam.es/~santacru>
Horario de atención al alumnado: Petición de cita previa por correo electrónico.

Dr. Alejandro Sierra

Departamento de Ingeniería Informática
Escuela Politécnica Superior
Despacho - Módulo: B-411 Edificio B - 4ª Planta
Teléfono: +34 91 497 2233
Correo electrónico: alejandro.sierra
Página web: <http://arantxa.ii.uam.es/~asierra/index.html>
Horario de atención al alumnado: Petición de cita previa por correo electrónico.

Ing. Alexandra Dumitrescu

Departamento de Ingeniería Informática
Escuela Politécnica Superior
Despacho - Módulo: B-402 Edificio B - 4ª Planta
Teléfono: +34 91 497 2293
Correo electrónico: doina.dumitrescu
Página web: <http://www.uam.es/doina.dumitrescu>
Horario de atención al alumnado: Petición de cita previa por correo electrónico.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

1.11. Objetivos del curso

Las **competencias comunes a la rama de Informática** que el estudiante adquiere con la asignatura Seminario-Taller de Software son:

- C13. Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los sistemas de información, incluidos los basados en web.

Los **resultados del aprendizaje** que el estudiante adquiere con la asignatura Seminario-Taller de Software son:

- Conocimiento a nivel de usuario (avanzado) y administración (intermedio) de distintos sistemas operativos.
- Capacidad para utilizar de manera eficaz y fluida entornos de programación, incluyendo herramientas de edición, compilación, ensamblaje y depurado de código.
- Capacidad para el diseño y realización de pruebas para comprobar la corrección, eficiencia, usabilidad y fiabilidad de un programa.
- Buenos usos de programación.
- Manejo de algunas aplicaciones informáticas especializadas.
- Manejo de herramientas de gestión de proyectos, gestión de versiones, trabajo colaborativo, etc.

A continuación, se especifican los objetivos generales y específicos de la asignatura Seminario-Taller de Software.

Los **objetivos generales** de STI-S son que los estudiantes al acabar el curso logren:

- G.1) Caracterizar la disciplina Informática.
- G.2) Describir la estructura y funcionamiento de las computadoras considerando sus dos componentes principales: hardware y software.
- G.3) Identificar, describir y relacionar las tareas que realiza un sistema operativo.
- G.4) Manejar el entorno UNIX.
- G.5) Identificar distintas actividades para el proceso de desarrollo de un sistema de software que conforman el proceso software.
- G.6) Diferenciar las visiones de gestión, técnica y de soporte del desarrollo de software.
- G.7) Aplicar técnicas de gestión de proyectos de software.
- G.8) Utilizar herramientas cooperativas como por ejemplo Google Docs.
- G.9) Utilizar herramientas de planificación de proyectos de software como por ejemplo Microsoft Project.
- G.10) Diseñar, codificar, depurar, probar, ejecutar e interpretar programas en el lenguaje de programación procedimental C.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

- G.11) Aplicar la modularización como técnica concreta para estructurar los programas.
- G.12) Utilizar entornos de programación en Linux.
- G.13) Mostrar interés e iniciativa para buscar, organizar y analizar críticamente información relevante como medio para el conocimiento y la toma de decisiones para lograr los objetivos especificados durante la implementación de las técnicas participativas y prácticas en grupo.
- G.14) Participar activamente en los análisis y discusiones de grupo que se establezcan al hilo del desarrollo del programa, cooperar con otros compañeros en el desarrollo de trabajos conjuntos y comunicar tanto en forma escrita como en forma oral con propiedad y corrección sus propias ideas y reflexiones así como los resultados de sus trabajos a los demás.

Los **objetivos específicos** de STI-S para cada unidad didáctica son que los estudiantes al acabar el curso deben ser capaces de:

UNIDAD 1: INTRODUCCIÓN A LA INFORMÁTICA

- 1.1) Definir los conceptos fundamentales de la Informática.
- 1.2) Describir la estructura y funcionamiento de los ordenadores.
- 1.3) Explicar el papel que desempeña el software en los ordenadores.

UNIDAD 2: MANEJO BÁSICO DE HERRAMIENTAS DE PROGRAMACIÓN

- 2.1) Definir el concepto de sistema operativo.
- 2.2) Explicar la evolución de los sistemas operativos.
- 2.3) Identificar y relacionar las tareas de los sistemas operativos.
- 2.4) Describir los comandos de uso más frecuente en UNIX.
- 2.5) Aplicar los comandos de uso más habitual del entorno UNIX.
- 2.6) Manejar un entorno de programación + Emacs/Vi (editor) + gcc + make (compilador) + ddd (depurador) en Linux para C.

UNIDAD 3: HERRAMIENTAS DE PLANIFICACIÓN Y SOPORTE DE TRABAJO EN EQUIPO

- 3.1) Explicar los conceptos básicos de la Ingeniería del Software.
- 3.2) Diferenciar entre las actividades de gestión, técnicas y de soporte para el éxito de un proyecto de software.
- 3.3) Describir las actividades principales de gestión de proyectos de software.
- 3.4) Aplicar algunas técnicas de planificación de proyectos de software sencillos.
- 3.5) Manejar el programa Microsoft Project y las herramientas Google Docs para preparar y planificar proyectos sencillos.
- 3.6) Desarrollar un talante negociador de gestión de proyectos de software.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

UNIDAD 4: MANEJO AVANZADO DE HERRAMIENTAS DE PROGRAMACIÓN

- 4.1) Interpretar el funcionamiento de programas que incluyan estructuras, definición de tipos, funciones, punteros, tablas, gestión dinámica de memoria y organización del código C en un entorno de programación.
- 4.2) Utilizar herramientas de gestión dinámica de memoria.
- 4.3) Programar, ejecutar, interpretar y probar programas en el lenguaje de programación procedimental C en un entorno de programación.
- 4.4) Documentar sistemas de software en el lenguaje de programación C.
- 4.5) Aplicar la programación estructurada y modular mediante el lenguaje de programación C.
- 4.6) Desarrollar un talante negociador fundamental para el desarrollo de un sistema de software.

1.12. Contenidos del programa

PROGRAMA SINTÉTICO

El enfoque de esta asignatura es de ingeniería, basada en el manejo de herramientas de gestión de proyectos de software y de implementación de sistemas de software. Asimismo, la participación de los estudiantes en forma activa es esencial, pues la planificación no sólo se debe analizar, sino también ponerse en práctica. Se pretende, con esta asignatura no solo transmitir conocimientos sino también desarrollar la capacidad de resolver problemas con más rapidez y mejor de lo que puede hacerlo hasta el momento el estudiante, fomentar el intercambio de ideas a través del trabajo cooperativo, mejorar la motivación del estudiante hacia su aprendizaje y favorecer la interacción social para el manejo de herramientas de programación y de gestión de proyectos, el desarrollo de sistemas de software y la resolución de problemas. Cabe destacar, que las unidades que componen la asignatura no son estrictamente secuenciales, sino que se relacionan, se retroalimentan unas a otras y se pueden realizar en distinto orden.

En concreto, la asignatura se ha estructurado en las siguientes cuatro unidades:

UNIDAD 1. INTRODUCCIÓN A LA INFORMÁTICA.

UNIDAD 2. MANEJO BÁSICO DE HERRAMIENTAS DE PROGRAMACIÓN.

UNIDAD 3. HERRAMIENTAS DE PLANIFICACIÓN Y SOPORTE DE TRABAJO EN EQUIPO.

UNIDAD 4. MANEJO AVANZADO DE HERRAMIENTAS DE PROGRAMACIÓN.

PROGRAMA DETALLADO

Para el conjunto de cuatro unidades se han definido los temas involucrados en el proceso de aprendizaje. Esta estructura de la asignatura Seminario-Taller de Software se muestra a continuación.

1. INTRODUCCIÓN A LA INFORMÁTICA

- 1.1. Conceptos básicos
- 1.2. Estructura y funcionamiento de los ordenadores
- 1.3. Software de un ordenador



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

2. MANEJO BÁSICO DE HERRAMIENTAS DE PROGRAMACIÓN

- 2.1. Definición de sistema operativo
- 2.2. Sistemas operativos de mayor difusión (Windows, UNIX y Linux)
- 2.3. Tareas de un sistema operativo: Gestión del procesador, gestión de la memoria, gestión de E/S y gestión de ficheros
- 2.4. Entorno UNIX y Linux
- 2.5. Entornos de programación en Linux

3. HERRAMIENTAS DE PLANIFICACIÓN Y SOPORTE DE TRABAJO EN EQUIPO

- 3.1. Ingeniería del Software: Definición y características
- 3.2. Modelo de ciclo de vida del software
- 3.3. Gestión de proyectos de software
- 3.4. Herramientas de gestión de proyectos

4. MANEJO AVANZADO DE HERRAMIENTAS DE PROGRAMACIÓN

- 4.1. Herramientas de manejo dinámico de memoria
- 4.2. Documentación de sistemas de software
- 4.3. Estructuración de programas y programación modular

1.13. Referencias de consulta

Los recursos para el aprendizaje por unidad se detallan a continuación. Se distinguen entre lecturas obligatorias y lecturas recomendadas.

UNIDAD 1: INTRODUCCIÓN A LA INFORMÁTICA

Lecturas Obligatorias:

- A. PRIETO, A. LLORIS, J. C. TORRES. *Introducción a la Informática*. 4ª ed. (McGraw Hill, 2006). Cap. 1: Introducción.
- Transparencias de la asignatura que se pueden encontrar en reprografía estando también disponibles en:
<http://www.uam-virtual.es/> .

Lecturas Recomendadas:

- J. G. BROOKSHEAR. *Computer Science: An Overview*. 9ª ed. (Pearson International Edition, 2007).

UNIDAD 2: MANEJO BÁSICO DE HERRAMIENTAS DE PROGRAMACIÓN

Lecturas Obligatorias:

- J. GARCÍA de JALÓN, I. AGUINAGA, A. MORA. *Aprenda LINUX como Si Estuviera en Primero*. (Universidad de Navarra, 2000).
- Transparencias de la asignatura que se pueden encontrar en reprografía estando también disponibles en:
<http://www.uam-virtual.es/> .



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

- Tutorial sobre el entorno de programación en Linux (editor de texto, compilador y depurador) para C que se puede encontrar en reprografía estando también disponible en:
<http://www.uam-virtual.es/> .

Lecturas Recomendadas:

- A. PRIETO, A. LLORIS, J. C. TORRES. *Introducción a la Informática*. 4ª ed. (McGraw Hill, 2006). Cap. 13: Sistemas Operativos.
- B. W. KERNIGHAN. *El Entorno de Programación UNIX*. (Prentice Hall Hispanoamericana, 1987).
- L. CEARRA. Sistemas Abiertos - Unix. Departamento de Publicaciones, Escuela Universitaria de Informática. (Universidad Politécnica de Madrid, 2000).
- Tutorial sobre el entorno de programación Visual en Windows (editor de texto, compilador y depurador) para C que se puede encontrar en:
<http://www.uam-virtual.es/> .

UNIDAD 3: HERRAMIENTAS DE PLANIFICACIÓN Y SOPORTE DE TRABAJO EN EQUIPO

Lecturas Obligatorias:

- R. S. PRESSMAN. *Ingeniería del Software. Un Enfoque Práctico*. 5ª ed. (McGraw Hill, 2001). Capítulo 2: El Proceso.
- Transparencias de la asignatura que se pueden encontrar en reprografía estando también disponibles en:
<http://www.uam-virtual.es/> .
- Tutorial sobre la herramienta de planificación Microsoft Project que se puede encontrar en reprografía estando también disponible en:
<http://www.uam-virtual.es/> .
- Tutorial sobre las herramientas Google Docs que se puede encontrar en reprografía estando también disponible en:
<http://www.uam-virtual.es/> .

Lecturas Recomendadas:

- R. S. PRESSMAN. *Ingeniería del Software. Un Enfoque Práctico*. 5ª ed. (McGraw Hill, 2001). Capítulo 1: El Producto; Capítulo 7: Planificación Temporal y Seguimiento del Producto.
- I. SOMMERVILLE. *Ingeniería de Software*. 6ª Edición. (Addison Wesley, 2002). Capítulo 1: Introducción; Capítulo 3: Procesos del Software; Apartados 4.1-4.3 (planificación).

UNIDAD 4: MANEJO AVANZADO DE HERRAMIENTAS DE PROGRAMACIÓN

Lecturas Obligatorias:

- B. W. KERNIGHAN, D. RITCHIE, *The C Programming Language*. 2nd ed. (Prentice Hall, 1988). Cap. 4: Funciones y la estructura del programa.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

- Tutorial sobre herramientas de gestión de memoria dinámica que se puede encontrar en reprografía estando también disponible en:
<http://www.uam-virtual.es/> .

Lecturas Recomendadas:

- B. W. KERNIGHAN, D. RITCHIE, *The C Programming Language*. 2nd ed. (Prentice Hall, 1988). Cap. 5: Apuntadores y arreglos; Cap. 6: Estructuras.
- B. W. KERNIGHAN. *La Práctica de la Programación*. (Pearson Educación, 2000).
- H. SCHILDT. *C: Guía de Autoenseñanza*. (Osborne/McGraw Hill, 2001).
- A. SIERRA URRECHO. *Programación en C/C++*. (Anaya Multimedia, 2005).
- Enlace de interés:
- [Librerías ANSI C](#)

2. Métodos docentes

En la asignatura Seminario-Taller de Software se utilizarán métodos y técnicas de aprendizaje cooperativo. En este apartado se realiza una conceptualización del aprendizaje cooperativo y se describen las actividades del estudiante y la implementación de los métodos y técnicas que se aplicarán en STI-S.

CARACTERÍSTICAS DEL APRENDIZAJE COOPERATIVO

El aprendizaje cooperativo es uno de los procedimientos que ha provocado mayores expectativas para resolver diferentes problemas en los contextos educativos. Según Slavin [7] la aplicación de los métodos de aprendizaje cooperativo atrae las miradas especialmente para resolver algunas cuestiones como: la mejora del rendimiento, la motivación, las relaciones interpersonales, el desarrollo de destrezas de pensamiento, y el incremento de las destrezas de colaboración. De hecho las investigaciones más actuales se dirigen al estudio de la relación entre el aprendizaje cooperativo y la motivación, al análisis de cómo la utilización de las técnicas de aprendizaje cooperativo constituyen un método adecuado para adquirir habilidades y competencias sociales [2][5].

El aprendizaje cooperativo se refiere a un conjunto de métodos instruccionales en donde los estudiantes trabajan en pequeños grupos (de tres a seis compañeros), generalmente heterogéneos en rendimiento. Este aprendizaje permite organizar el propio pensamiento y fomenta el contraste de diferentes puntos de vistas, revisando tanto las ideas propias como las ideas de los restantes miembros del equipo. Los integrantes de cada grupo son responsables no sólo de aprender el material de clase, sino de ayudar a que todo su grupo lo aprenda. Por último se recompensa el rendimiento obtenido como consecuencia del trabajo en grupo [1][6]. Una definición que concreta esos aspectos es la que proponen Johnson y Johnson [3]. Definen al *aprendizaje cooperativo* como aquella situación de aprendizaje en la que los objetivos de los participantes se hallan estrechamente vinculados, de tal manera que cada uno de ellos "sólo puede alcanzar sus objetivos si y sólo si los demás consiguen alcanzar los suyos".



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

Por eso, agrupar a los estudiantes no es en sí mismo un factor que incite al trabajo colectivo. Slavin [6] insiste en dos condiciones fundamentales: por un lado, el trabajo debe estar dirigido a conseguir metas de grupo; por otro lado, el éxito en conseguir tales metas depende del aprendizaje individual de todos los miembros del grupo.

Entre los elementos que constituyen el método de aprendizaje cooperativo se pueden considerar los siguientes [4]: la interdependencia positiva, la interacción cara a cara, dar responsabilidad a cada estudiante del grupo, desarrollar las habilidades del grupo y las relaciones interpersonales y la reflexión sobre el trabajo del grupo. También, los autores señalan las principales características del trabajo cooperativo que consisten en: un equipo cooperativo, una administración a través de reglas, voluntad de cooperar, habilidad para cooperar y escuchar a los demás, resolver problemas y apoyarse mutuamente y estructura y roles.

Para evaluar, hay que tener en cuenta, al menos, los siguientes aspectos: a) objetivos, b) niveles de cooperación, c) esquema de interacción y d) evaluación de resultados. El profesor ha de cuidar que los elementos de carácter psicosocial estén siempre presentes en el trabajo cooperativo, es decir, que se reflejen en: el buen funcionamiento interpersonal en grupo, la interacción cara a cara entre los miembros (de tres a seis personas), la responsabilidad individual para aumentar la comprensión y la interdependencia positiva a través de objetivos comunes, de la división de los materiales, los recursos y la información, y de la asignación de roles.

REFERENCIAS

- [1] Díaz-Aguado, M. J. (2003). *Educación Intercultural y Aprendizaje Cooperativo*. Madrid: Pirámide.
- [2] Echeíta, G. (1995). *El Aprendizaje Cooperativo. Un Análisis Psicosocial de sus Ventajas Respecto a Otras Estructuras de Aprendizaje*. En P. Fernández y A. Melero (Comps.), *La Interacción Social en Contextos Educativos*, 167-189. Madrid: Siglo XXI.
- [3] Johnson, D. W., y Johnson, R. (1992). *Positive Interdependence: Key to Effective Cooperation*. En R. Hertz-Lazarowitz, N. Miller (Eds.), *Interaction in Cooperative Groups*, 174-199. Cambridge, Mass: Cambridge University Press.
- [4] Johnson, D. W., Johnson, R., y Holubec, E. J. (1993). *Cooperation in the Classroom*. 6ª ed. Edina, MN: Interaction Book Company.
- [5] León, B., Gonzalo, M., y Vicente, F. (2004). El entrenamiento previo en habilidades sociales como factor mediador en la eficacia del aprendizaje cooperativo. *Actas del IV Congreso Internacional de Psicología y Educación: Calidad Educativa*, 1604-1614, Almería, 30-31 de Marzo y 1-2 de Abril.
- [6] Slavin, R. E. (1987). Cooperative learning and the cooperative school. *Educational Leadership*, 45(3), 7-15.
- [7] Slavin, R. E. (1991). Synthesis of research on cooperative learning. *Educational Leadership*, 48(5), 71-82.

ACTIVIDADES DEL ESTUDIANTE. MÉTODOS Y TÉCNICAS A APLICAR

Para alcanzar los objetivos y capacidades previstos (véase apartado 1.11), la asignatura consta de sesiones prácticas en laboratorios con ordenadores, donde gran parte del trabajo se organiza en clases de tipo participativo. El papel de las



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

técnicas participativas es servir de vehículo para el desarrollo de problemas/trabajos en forma conjunta, donde se fomenta el diálogo y el trabajo cooperativo, tutorizado por el profesor, y también las discusiones de alto nivel de las soluciones, de los métodos de solución y problemas de cada trabajo.

A lo largo del curso los estudiantes trabajarán en pequeños grupos (3-4 miembros) de aprendizaje cooperativo. Se les brindará una guía para la elaboración de memorias de las prácticas. Asimismo, se les facilitará ayudas para el aprendizaje de herramientas, incluyendo tutoriales y vídeos, y se les ofrecerá a los estudiantes tutorías para mejorar la resolución de problemas computacionales.

A continuación, se describen las tareas y trabajos que se realizarán en cada una de las unidades involucradas en STI-S, describiendo los métodos y técnicas participativos y cooperativos. Estas actividades requieren que el estudiante asista a las clases y participe en el desarrollo de las mismas conforme a las dinámicas de trabajo que en cada caso se establezcan. La Tabla 4, en el apartado 5, muestra una posible distribución en el tiempo de estas actividades.

Para la Unidad 1: *Introducción a la Informática*, se aplicará la siguiente técnica de grupo en las sesiones de prácticas en equipos de 3-4 integrantes:

- Técnica de grupo de articulación y reflexión sobre ¿qué es Informática? Puesta en común en la pizarra sobre las definiciones que cada equipo elabora, comparativa entre las definiciones apuntadas, considerando elementos comunes y no comunes, y establecimiento de una definición adecuada de la disciplina Informática. Carga horaria: 1 sesión de 2 horas.

Para la Unidad 2: *Manejo Básico de Herramientas de Programación*, se aplicarán las siguientes técnicas de grupo en las sesiones de prácticas en equipos de 3-4 integrantes:

- Prácticas de laboratorio en ordenadores. Carga horaria presencial: 6 sesiones de 2 horas que hacen un total de 12 horas de laboratorio con la presencia del profesor para el desarrollo de 2 prácticas sobre el entorno UNIX, el sistema operativo LINUX y herramientas de programación en LINUX, a saber "Aprendiendo Fundamentos de Linux" y "Aprendiendo a Programar en Linux". En estas sesiones, se practicarán varios comandos Linux, y se estudiarán herramientas de programación en Linux como: editores de código, Emacs o Vi, el compilador gcc y el depurador ddd. Los equipos conformados deberán presentar una memoria sobre la Práctica 1 y realizar una prueba procedimental en el laboratorio sobre la Práctica 2.

Para la Unidad 3: *Herramientas de Planificación y Soporte de Trabajo en Equipo* se realizará la aplicación de las siguientes técnicas de grupo en los equipos de 3-4 integrantes:

- Técnica de grupo de articulación y reflexión sobre ¿qué es Ingeniería del Software y qué herramientas existen para gestionar proyectos? Puesta en común en la pizarra sobre las definiciones que cada equipo elabora y las herramientas de gestión enumeradas, comparativa entre las definiciones



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

apuntadas, considerando elementos comunes y no comunes, y establecimiento de una definición adecuada de Ingeniería del Software así como también de una clasificación de las herramientas de gestión seleccionadas. Carga horaria: 1 sesión de 2 horas.

- Prácticas de laboratorio en ordenadores. Carga horaria presencial: 2 sesiones de 2 horas que hacen un total de 4 horas de laboratorio con la presencia del profesor para el desarrollo de 1 práctica sobre el entorno de planificación de proyectos Microsoft Project y las herramientas web Google Docs, denominada "Aprendiendo a Organizar y Planificar Proyectos de Software". En esta práctica se preparará y planificará el desarrollo del problema computacional que se implementará en la práctica correspondiente a la Unidad 4. Los equipos conformados deben presentar una memoria sobre la práctica realizada.

Para la Unidad 4: *Manejo Avanzado de Herramientas de Programación* se aprenderán herramientas de gestión dinámica de memoria y se integrará la aplicación de estas herramientas con las herramientas aprendidas en los temas previos (editores, gcc, make, ddd, etc.) junto con la elaboración de la documentación correspondiente para resolver un problema de programación que requiera el uso de punteros y gestión dinámica de memoria, en los equipos de 3-4 integrantes durante un total de 6 horas de laboratorio con la presencia del profesor en 3 sesiones de 2 horas cada una. Esta práctica se titula "Aprendiendo a Usar Herramientas de Gestión de Memoria Dinámica". Los equipos conformados deben presentar una memoria sobre la práctica realizada y realizar una exposición oral de los resultados obtenidos en las Prácticas 3 y 4 de 10 minutos de duración y 5 minutos de preguntas y respuestas.

Durante estas sesiones participativas todos los entregables serán oportunamente devueltos a cada equipo con las retroalimentaciones correspondientes y las valoraciones pertinentes.

Además, en la semana previa a la entrega de las Prácticas 1 y 4 se realizan tutorías programadas, es decir sesiones con todos los equipos para seguimiento y corrección de trabajos. Las tutorías programadas se reflejan en el cronograma del curso (Tabla 4). Para la Práctica 1, en esta sesión se realizará una prueba de opción múltiple a fin de evaluar el rendimiento individual y para la Práctica 4 se evaluarán las presentaciones orales por equipo sobre las Prácticas 3 y 4 (Tabla 4).

3. Tiempo de trabajo del estudiante

Se asignan 25 horas de trabajo a cada crédito europeo, por tanto una asignatura de 3 créditos conlleva 75 horas de trabajo del estudiante, que incluyen tanto tareas presenciales como no presenciales. En la Tabla 1 se indica el porcentaje de cada actividad respecto al total de 75 horas para los estudiantes que han seleccionado el método de evaluación continua con asistencia obligatoria a clase en relación con las directrices del Espacio Europeo de Educación Superior.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

		Nº de horas	Porcentaje
Presencial	Clases teóricas	0 h (0%)	45% = 34 horas
	Clases prácticas	28 h (37%)	
	Tutorías programadas a lo largo del semestre	3 h (4%)	
	Realización de exámenes y/o presentaciones	3 h (4%)	
No presencial	Realización de actividades prácticas	18 h (24%)	55% = 41 horas
	Estudio semanal (0,5 h x 14 semanas)	7 h (10%)	
	Preparación de exámenes y/o presentaciones	16 h (21%)	
Carga total de horas de trabajo: 25 horas x 3 ECTS		75 h	

Tabla 1. Distribución del Tiempo de Trabajo del Estudiante en la Asignatura Seminario-Taller de Software

4. Métodos de evaluación y porcentaje en la calificación final

PONDERACIÓN

La evaluación del estudiante se realizará midiendo su nivel de consecución de objetivos. Esta evaluación será ponderada en dos dimensiones generales, vinculadas a las distintas competencias establecidas como objetivos:

- Asimilación de los contenidos y procedimientos: Hasta 9 puntos.
- Participación e iniciativa, organización del trabajo y presentaciones: Hasta 1 punto.

CRITERIOS DE EVALUACIÓN

Los criterios de evaluación en base a los cuales se valorarán las dimensiones establecidas serán los siguientes:

- Asimilación de los contenidos

Criterios:

¿Es capaz de indicar los principales contenidos que se han analizado en la asignatura?

¿Es capaz de establecer relaciones entre las ideas y los conceptos estudiados?

¿Ha desarrollado un criterio propio respecto a los conceptos/métodos/técnicas/herramientas analizados?

¿Es capaz de evaluar los efectos de sus decisiones?

¿Ha sacado conclusiones propias de las lecturas que haya realizado de modo independiente?

¿Se hace una idea de lo que desconoce con vistas a seguir aprendiendo e indagando?



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

Estas competencias se evaluarán a través de la calidad de los trabajos en grupo que se realicen a lo largo del curso, así como a través de entrevistas sobre las prácticas de la asignatura.

- Participación e iniciativa, organización del trabajo y presentaciones

Criterios

¿Ha asistido regularmente a clase?

¿Ha realizado todos los trabajos de grupo que se han pedido?

¿Participa en clase preguntando, dando opiniones o respondiendo a preguntas del profesor?

¿Ha completado sus conocimientos con lecturas no obligatorias o de actividades?

¿Ha mostrado actitud colaboradora y respetuosa hacia los compañeros?

¿Ha realizado la memoria de las prácticas conforme a las normas y criterios establecidos?

¿Sus trabajos están cuidados con respecto a su presentación?

¿Su carpeta con la documentación y los apuntes está ordenada y cuidada?

¿Es capaz de expresarse oralmente de forma clara y con precisión?

¿Es capaz de argumentar y defender sus ideas ante los demás?

Estas competencias se evaluarán a través de: a) la observación del profesor, el registro de asistencia y la entrega de trabajos obligatorios y voluntarios; b) el análisis de la propia carpeta personal y su contenido; y c) las intervenciones que realice en clase como motivo de preguntas o respuestas y con las presentaciones que realice.

FORMAS DE EVALUACIÓN DE LA ASIGNATURA

En la Tabla 2 se muestran las fórmulas correspondientes a la nota final de la asignatura Seminario-Taller de Software tanto para el método de evaluación continua con asistencia obligatoria a clase como para el método de evaluación no continua sin asistencia obligatoria a clase.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

NOTA FINAL (NF) DE STI-S - Evaluación continua con asistencia obligatoria a clase	$20\% \times P1 + 25\% \times P2 + 25\% \times P3 + 30\% \times P4$
	<ul style="list-style-type: none">• P1 = Práctica 1• P2 = Práctica 2• P3 = Práctica 3• P4 = Práctica 4 Valor mínimo exigido de P1, P2, P3 y P4 para el cálculo anterior: 5. En caso contrario, NF será igual a: ($20\% \times \text{Mínimo}(5, P1) + 25\% \times \text{Mínimo}(5, P2) + 25\% \times \text{Mínimo}(5, P3) + 30\% \times \text{Mínimo}(5, P4)$).
NOTA FINAL (NF) DE STI-S - Evaluación no continua sin asistencia obligatoria a clase	$60\% \times \text{EFP} + 40\% \times (20\% \times P1 + 25\% \times P2 + 25\% \times P3 + 30\% \times P4)$
	<ul style="list-style-type: none">• EFP = Nota del Examen Final de Prácticas• P1 = Práctica 1• P2 = Práctica 2• P3 = Práctica 3• P4 = Práctica 4 Valor mínimo exigido de EFP, P1, P2, P3 y P4 para el cálculo anterior: 5. En caso contrario, NF será igual a: ($60\% \times \text{Mínimo}(5, \text{EFP}) + 40\% \times \text{Mínimo}(5, (20\% \times \text{Mínimo}(5, P1) + 25\% \times \text{Mínimo}(5, P2) + 25\% \times \text{Mínimo}(5, P3) + 30\% \times \text{Mínimo}(5, P4)))$).

Tabla 2. Sistema de Evaluación de la Asignatura Seminario-Taller de Software

Es necesario aprobar con una nota mínima de 5 cada una de las cuatro prácticas. Cada práctica se evaluará como 9 (asimilación de los contenidos) + 1 (participación) según la ponderación mencionada anteriormente. La nota final de la asignatura se obtiene mediante la siguiente fórmula ponderada de las cuatro prácticas para los estudiantes que optaron por evaluación continua:

$$20\% \times P1 + 25\% \times P2 + 25\% \times P3 + 30\% \times P4$$

o mediante el 40% de la nota ponderada de las cuatro prácticas y el 60% del examen final que involucra todos los temas desarrollados en dichas prácticas realizado por los estudiantes que no optan por evaluación continua:

$$60\% \text{EFP} + 40\% (20\% \times P1 + 25\% \times P2 + 25\% \times P3 + 30\% \times P4)$$

Para la nota de la Práctica 1 se hará media entre la nota grupal de la práctica y la nota individual de la prueba de opción múltiple. La nota de la Práctica 2 será la evaluación procedimental que se llevará a cabo en forma grupal e individual. Para la nota de la Práctica 4 se hará media entre la nota grupal de la práctica y la nota individual de la exposición oral.

El Examen Final de Prácticas es sólo para los estudiantes que han seleccionado el método de evaluación no continua y se realizará en el mes de enero de 2012. Estos estudiantes deben estar presentes en las sesiones de entrega de cada práctica para su evaluación oral parcial correspondiente sobre la práctica realizada.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

Por una parte, se considera que el estudiante se ha presentado a la convocatoria correspondiente si presenta al menos dos prácticas. Es decir, el número mínimo de pruebas a las que el estudiante se ha de presentar para recibir una calificación numérica es 2. Por debajo de este número el estudiante recibirá la calificación "No evaluado". Aunque el estudiante no se presente a la prueba final, siempre que se haya presentado a este mínimo de pruebas, recibirá una calificación numérica. Por otra parte, si un estudiante pasa del método de evaluación continua al método de evaluación no continua, podrá obtener un 5 como máximo en su calificación final.

En relación con la evaluación extraordinaria para ambos métodos, los estudiantes deberán presentar todas las prácticas correctamente y presentarse al examen final de prácticas a desarrollarse en el mes de junio de 2012.

Nota: Seminario-Taller de Informática (STI) se aprueba con una nota mínima de 5. La nota de Seminario-Taller de Informática se obtendrá mediante la media de las notas obtenidas en Seminario-Taller de Software (STI-S) y Seminario-Taller de Hardware (STI-H). Es necesario obtener una nota mínima de 5 en cada una de las asignaturas, STI-S y STI-H, para poder hacer media. En caso contrario, la nota final de STI será igual a: $(50\% \times \text{Mínimo}(5, \text{STI-S}) + 50\% \times \text{Mínimo}(5, \text{STI-H}))$.

NORMAS

Los entregables se regirán por las siguientes normas:

1. Se propondrán cuatro prácticas a lo largo del curso, de entrega y/o evaluación obligatoria para superar la asignatura.
2. Todas las prácticas se puntuarán y su nota se tendrá en cuenta en la nota final, como se ha indicado anteriormente.
3. Es necesario tener al menos un 5 en todas las notas prácticas para poder aplicar la fórmula ponderada anterior.
4. Para aprobar las prácticas será necesario:
 - 4.1. Entregar una memoria de las Prácticas 1, 3 y 4 del curso, según se especifica después en este documento.
 - 4.2. Realizar la evaluación de rendimiento individual de la Práctica 1.
 - 4.3. Realizar la evaluación procedimental grupal e individual de la Práctica 2.
 - 4.4. Entregar el código fuente correspondiente a los problemas de la Práctica 4. La misma se evaluará también con respecto a:
 - Corrección funcional: respuesta correcta a datos de entrada, ya sean normales, excepcionales o incorrectos.
 - Organización del código y calidad del mismo.
 - 4.5. Presentación oral de las Prácticas 3 y 4.
 - 4.6. Las prácticas se evaluarán en una escala del 0 al 10.
5. En el enunciado de las prácticas en la Web estarán publicadas las fechas de entrega de éstas.
6. Toda práctica entregada con el requisito de desarrollo de software que no compile o con errores de ejecución se penalizará un 20% de la nota final de prácticas.
7. Una práctica no entregada supone un suspenso en las prácticas.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

8. Las prácticas copiadas suponen un suspenso en la asignatura tanto para el equipo copiado como para el copiadador.
9. La evaluación de las prácticas puede requerir la realización de una entrevista por cada equipo donde se evaluará la realización de la práctica, así como sobre los conocimientos de la asignatura que han sido tenidos en cuenta en su realización.
10. El plazo de entrega de prácticas será estricto, y siempre se realizará en el momento establecido, que normalmente será al comienzo de la clase de la práctica siguiente a la que se entrega. Si no se entrega en ese momento, se considerará entregada un día después.
11. Por cada día de retraso en la entrega se penalizará un 10% de la nota final de prácticas.
12. Las prácticas se realizarán en equipos de 3-4 personas que dispondrán de un turno reservado de dos horas a la semana.
13. No se aceptarán prácticas realizadas individualmente, ni cambios de equipo una vez entregada la primera práctica.
14. La composición de los equipos se establecerá el primer día de prácticas y después no podrá modificarse, salvo circunstancias excepcionales.
15. Cada equipo debe estar apuntado en su turno de prácticas hasta tres días luego de la primera sesión de clases de STI-S (que comienza en la semana del 12 de septiembre de 2011, en el día correspondiente a cada grupo de prácticas). Se habilitará un registro de equipos de prácticas para tal fin, que se concluirá hasta tres días luego del comienzo de las clases de cada grupo de STI-S a las 20:00 horas. **ATENCIÓN:** Todos los estudiantes deben registrarse independientemente de que tengan equipo o no, ya que si no tienen equipo, el profesor respectivo les empareja con alguien que no tenga equipo en el turno correspondiente.
16. Los enunciados de las prácticas estarán disponibles en reprografía tres días antes de su fecha de comienzo. También podrán ser consultados a través de la página Web de la asignatura.
17. Las notas asignadas a los integrantes de un mismo equipo por una misma práctica podrán ser diferentes cuando el profesor de prácticas encuentre justificación para ello.

MEMORIA

La memoria de cada práctica deberá incluir:

1. Nombre de los estudiantes, número y nombre del equipo, turno, fecha y número de práctica.
2. Descripción del problema o práctica en cuestión.
3. Análisis para la resolución del problema planteado.
4. Aspectos de interés en la realización de la práctica a tener en cuenta por el profesor.
5. En su caso, respuesta a las preguntas que se planteen en el enunciado de la práctica y/o material solicitado.
6. Discusión y razonamiento de los resultados obtenidos.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

ENTREGAS

Serán enviados por medio de la Web de la asignatura para entrega de prácticas los ficheros generados en la realización de la práctica, y entregada en el buzón de la asignatura una memoria sobre el desarrollo realizado.

La entrega consistirá en un fichero comprimido (.zip), en el que se incluirá, para la Práctica 4 realizada, lo siguiente:

- Código fuente, debidamente comentado, de todos los programas realizados en la práctica (ficheros .c).
- Ficheros de cabecera (ficheros .h).
- Fichero Makefile.
- Ficheros de datos necesarios para la ejecución del programa, si los hay.

No se entregarán, por tanto, los ficheros ejecutables, con el código objeto (*.o), ni los ficheros de los proyectos y los entornos de trabajo. Dentro del fichero comprimido, cada fichero tendrá un nombre significativo que indique su función. Las extensiones dadas a los nombres de los ficheros serán las usuales: c para los fuentes, h para los ficheros de cabecera, etc.

En el caso de entrega electrónica de prácticas, el nombre del fichero de entrega debe ser *PxTyEz.zip* donde x es el número de práctica, y el turno correspondiente y z el número de equipo dentro del turno. Por ejemplo, si el equipo 6 del turno 164 quiere entregar la Práctica 3 debería enviar un fichero con el nombre *P3T164E6.zip*.

Se entregarán las prácticas desde el Sistema de Entrega de Prácticas de la plataforma Moodle (<http://uam-virtual.es>) de la Escuela Politécnica Superior donde se estructura STI-S.

PRÁCTICAS CURSO 2011/2012

En el presente curso se contempla la realización de cuatro prácticas. Los temas de las prácticas, así como la fecha de la semana de su evaluación y/o entrega, en lo posible el día correspondiente a cada grupo de STI-S, están en la Tabla 3.



Asignatura: Seminario-Taller de Software
Código: 17818
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Obligatoria
Nº de créditos: 3 ECTS

PRÁCTICA	TEMA	SEMANA DE PUBLICACIÓN	SEMANA DE EVALUACIÓN / [SEMANA DE ENTREGA]	Nº DE SEMANAS
1	Aprendiendo Fundamentos de Linux	12/09/2011	10-14/10/2011 [17-21/10/2011]	3
2	Aprendiendo a Programar en Linux	10/10/2011	31/10-4/11/2011	3
3	Aprendiendo a Organizar y Planificar Proyectos de Software	31/10/2011	[28/11-2/12/2011]	3
4	Aprendiendo a Usar Herramientas de Gestión de Memoria Dinámica	21/11/2011	12-16/12/2011 [19/12/2011]	3

Tabla 3. Temas y Entregas de las Prácticas de la Asignatura Seminario-Taller de Software

Sobre las tutorías: Cada estudiante se dirigirá al profesor de prácticas para las dudas que le puedan surgir fuera del horario de laboratorio. Para ello primero se requiere concretar cita con el profesor vía e-mail o respetar los días y horarios predeterminados.

5. Cronograma

El cronograma del estudiante que sigue el método de evaluación con asistencia obligatoria a clase se muestra en la Tabla 4. Este cronograma tiene carácter orientativo y le ofrece al estudiante una visión global del curso y le permite planificar su trabajo de forma realista.



Asignatura: Seminario-Taller de Software
 Código: 17818
 Centro: Escuela Politécnica Superior
 Titulación: Grado en Ingeniería Informática
 Nivel: Grado
 Tipo: Obligatoria
 N° de créditos: 3 ECTS

Semana	Contenido	Horas presenciales	Horas no presenciales
1	- Presentación y motivación de la asignatura, su normativa y dividir a los estudiantes en equipos de 3-4 integrantes. - Unidad 1 Introducción a la Informática , con una puesta en común en la pizarra sobre las definiciones de Informática elaboradas por cada equipo, considerando elementos comunes y no comunes, y estableciendo al final de la clase una definición adecuada de Informática.	2	2 (Trabajo del estudiante: - Lectura de material propuesto sobre Unidad 1)
2	- Comienzo de la Unidad 2 Manejo Básico de Herramientas de Programación , Práctica 1 "Aprendiendo Fundamentos de Linux", cada equipo debe seguir el guión proporcionado por el profesor y completar la práctica en 3 semanas.	2	2 (Trabajo del estudiante: - Lectura de material propuesto sobre Unidad 2 - Realización de Práctica 1)
3	- Sesión de laboratorio para seguir con la Práctica 1 y resolver dudas.	2	2 (Trabajo del estudiante: - Realización de Práctica 1)
4	- Sesión de laboratorio para terminar la Práctica 1.	2	2 (Trabajo del estudiante: - Realización de Práctica 1 - Preparación prueba de opción múltiple)
4	- Tutorías en equipo sobre la Práctica 1. - Prueba de opción múltiple individual sobre la Práctica 1.	1,5	
5	- Entrega de la Práctica 1, y explicación de la Práctica 2 "Aprendiendo a Programar en Linux" a completar en 3 sesiones, el profesor explicará cómo usar en Linux un entorno de programación, editar código con vi/emacs, compilar usando gcc y make, y depurar código usando ddd.	2	2 (Trabajo del estudiante: - Entrega de Práctica 1 - Realización de Práctica 2)
6	- Sesión de laboratorio para seguir con la Práctica 2 y resolver dudas.	2	2 (Trabajo del estudiante: - Realización de Práctica 2)
7	- Sesión de laboratorio para la evaluación de la Práctica 2.	2	2 (Trabajo del estudiante: - Realización de Práctica 2 - Preparación para la evaluación de Práctica 2)

Tabla 4. Cronograma de la Asignatura Seminario-Taller de Software



Asignatura: Seminario-Taller de Software
 Código: 17818
 Centro: Escuela Politécnica Superior
 Titulación: Grado en Ingeniería Informática
 Nivel: Grado
 Tipo: Obligatoria
 N° de créditos: 3 ECTS

Semana	Contenido	Horas presenciales	Horas no presenciales
8	- Comienzo de la Unidad 3 Herramientas de Planificación y Soporte de Trabajo en Equipo con una sesión inicial de puesta en común sobre qué es la Ingeniería del Software y qué herramientas existen para gestionar proyectos. Entrega del guión de la Práctica 3 "Aprendiendo a Organizar y Planificar Proyectos de Software" a completar en 3 sesiones.	2	2 (Trabajo del estudiante: - Entrega de Práctica 2 - Lectura de material propuesto sobre Unidad 3)
9	- Sesión de laboratorio para seguir con la Práctica 3 y resolver dudas.	2	2 (Trabajo del estudiante: - Realización de Práctica 3)
10	- Sesión de laboratorio para terminar la Práctica 3.	2	2 (Trabajo del estudiante: - Realización de Práctica 3)
11	- Entrega de la Práctica 3, y comienzo de la Unidad 4 Manejo Avanzado de Herramientas de Programación con el desarrollo de un tutorial sobre herramientas de gestión de memoria dinámica y la asignación a cada grupo de un problema computacional (Práctica 4 "Aprendiendo a Usar Herramientas de Gestión de Memoria Dinámica") que deben resolver utilizando estas herramientas y las herramientas de programación en Linux estudiadas, usando Google Docs para compartir información entre los miembros del grupo y planificando el proyecto adecuadamente con un diagrama de Gantt realizado en Microsoft Project. Esta práctica se completará en 3 sesiones.	2	2 (Trabajo del estudiante: - Entrega de Práctica 3 - Lectura de material propuesto sobre Unidad 4)
12	- Sesión de laboratorio para seguir con la Práctica 4 y resolver dudas.	2	2 (Trabajo del estudiante: - Realización de Práctica 4)
13	- Sesión de laboratorio para terminar la Práctica 4.	2	2 (Trabajo del estudiante: - Realización de Práctica 4 - Preparación exposición oral en equipo de las Prácticas 3 y 4)
14	- Tutorías en equipo sobre la Práctica 4. - Exposición oral de los resultados de aprendizaje obtenidos en las Prácticas 3 y 4 por equipo.	1,5	
14	- Sesión de laboratorio para seguir con la Práctica 4 y resolver dudas. - Entrega de la Práctica 4.	2	2 (Trabajo del estudiante: - Realización de Práctica 4 - Entrega de Práctica 4)

Tabla 4. Cronograma de la Asignatura Seminario-Taller de Software (Continuación)