



CC51A - Ingeniería de Software

Sergio Ochoa D.

Clase 1 - Versión 3.

Responsables:

Profesor a Cargo:

Sergio Ochoa D. sochoa@dcc.uchile.cl

Tel: + 56 2 678-4364/5

Oficina 313 (DCC).

Auxiliar:

Andrés Neyem aneyem@dcc.uchile.cl

Tel: + 56 2 678-4890

Ayudante:

Guillermo Poblete Hoffmann swordbug@vtr.net

Organización

1. Definición de Objetivos
2. Contenidos del Curso
3. Metodología de Desarrollo del Curso
4. Roles de los Alumnos
5. Requisitos de Aprobación
6. Evaluación
7. Protocolo de Comunicación.
8. Guía de Supervivencia

Objetivo

- ◆ Desarrollar un Sistema Real, apoyándose en un Estándar.
- ◆ Conocer un Estándar (ESA).
- ◆ Trabajar en Equipo.
- ◆ Desempeñar un Rol.
- ◆ Presentar el Nuevo Escenario de Desarrollo (Internet/Intranet).

Contenidos (1)

- 1. Introducción**
- 2. Gestión de Proyectos de Software**
- 3. Especificación de Requisitos**
- 4. Aseguramiento de Calidad de Software**
- 5. Diseño de Sistemas de Software**
- 6. Implementación de Sistemas de Software**
- 7. Entrega del Producto**

Metodología (1)

Cátedra:

- Clases Teóricas (B211 - Mar-Jue. 8:30-10:00hs).
- Clases Auxiliares (B213 - Lunes 14:30 a 16:00hs)

Alumnos:

- Selección y Ejecución de un Rol.
- Ejecución de un Proyecto.

Metodología (2)

1º- A cada alumno se le asigna un rol, de acuerdo a su Curriculum y a sus preferencias. Los roles posibles son: Administrador de Proyectos, Analista, Diseñador, Implementador, o Téster.

2º- Los alumnos son divididos en equipos de trabajo de 5 a 7 personas, y se les asignará un proyecto a desarrollar por ellos durante el semestre.

3º- Cada equipo ejecuta el proyecto según las pautas dadas por la cátedra.

Metodología (3)

- 4°- Habrá dos tipos de proyectos: *de implantación y extensión*, y *de desarrollo e implantación*.
- 5°- La asignación de proyectos a equipos de trabajo la hace el profesor.
- 6°- Al finalizar el semestre, cada equipo de trabajo, deberá entregar una versión ejecutable del sistema, ajustándose a los requisitos especificados por el cliente.

Roles (1)

Administrador de Proyecto: Es el coordinador y responsable del trabajo de su equipo. Entre sus responsabilidades está:

1. Coordinar el trabajo de los distintos miembros del equipo.
2. Interactuar con el Cliente.
3. Velar por el cumplimiento de los objetivos, plazos y costos comprometidos.
4. Planificar y Administrar el Proyecto de Software.

Este es uno de los roles más críticos dentro de cualquier proyecto de desarrollo de software.

Roles (2)

Analista: Es el encargado de especificar los requisitos del problema a desarrollar. Entre sus tareas está:

1. Entrevistar al cliente.
2. Generar los documentos de requisitos de software (SRD) y de usuario (URD), dentro de los plazos comprometidos.
3. Velar porque el diseño cumpla con los requisitos (funciones de téster).
4. Velar porque el producto final cumpla con los requisitos (funciones de téster).

Roles (3)

Diseñador: Es el encargado de generar el diseño del sistema. Entre sus funciones está:

1. Generar el diseño arquitectónico y diseño detallado del sistema (DD-Documento de Diseño), basándose en los requisitos.
2. Generar prototipos rápidos del sistema (con analistas e implementadores) para chequear los requisitos entregados por el cliente.
3. Velar porque el producto final se ajuste al diseño realizado (funciones de téster).

Roles (4)

Implementador: Es el encargado de implementar el sistema. Entre sus funciones está:

1. Implementar los diseños del sistema, especificados por los diseñadores. (BD. y código ejecutable)
2. Implementar los prototipos rápidos para chequear los requisitos.
3. Realizar la primera batería de pruebas.
4. Diseñar (junto al tester) y llevar a cabo el plan de implantación.

Roles (5)

Téster (Control de Calidad): Es el encargado de asegurar la calidad de cada uno de los productos (documentos, prototipos, etc). Entre sus tareas están:

1. Coordinar las revisiones de los productos del proyecto.
2. Velar por la adhesión al estándar adoptado para el desarrollo.
3. Velar por la completitud, y exactitud (no ambigüedades) de los documentos.
4. Velar por la calidad del producto final (cumplimiento de los requisitos).
5. Definir el plan de pruebas y de implantación junto con el implementador.

Requisitos de Aprobación

Para aprobar el curso, se requiere:

- Promedio de notas de los controles debe ser mayor o igual a 4.0 (incluyendo el examen, y el reemplazo de notas).
- Promedio de notas del proyecto debe ser mayor o igual a 4.0.
- La nota del examen también debe ser mayor o igual a 4.0.

$$NF = 0.6 * PNP + 0.4 * PNC$$

NF = Nota Final

PNP = Promedio de Notas del Proyecto

PNC = Promedio de Notas de Controles

Evaluación (1)

60 % - Proyecto:

- 10 % corresponde a la nota del ***análisis***.
- 10 % corresponde a la nota del ***diseño***.
- 10 % corresponde a la nota de la ***implementación***.
- 10 % corresponde a la nota de la ***implantación y entrega***.
- 20 % corresponde a la nota del ***profesor***, según la capacidad mostrada para desempeñar el rol asignado.

Evaluación (2)

40 % - Controles:

- 15 % corresponde a la nota del examen.
- 12.5 % corresponde a la nota del control 1.
- 12.5 % corresponde a la nota del control 2.

La nota del examen puede reemplazar a la de uno de los controles.

Evaluación (3)

NOTA:

- 1- Si más del 50 % de los miembros del equipo de trabajo está de acuerdo con expulsar a uno de sus integrantes, puede hacerlo.
- 2- Para ello deberá contar con el consentimiento del profesor.
- 3- El expulsado tendrá un 1 como nota final del proyecto, y reprobará el curso.

Evaluación (4)

NOTA:

- 1- El administrador de cada proyecto puede recomendar al profesor, la expulsión justificada de un miembro del equipo de trabajo, sin necesidad de que la mayoría esté de acuerdo.
- 2- Si el alumno es expulsado, la responsabilidad será compartida entre el administrador del proyecto y el profesor.
- 3- El expulsado tendrá un 1 como nota final del proyecto, y reprobará el curso.

Bibliografía

- “*Software Engineering - A Practitioner’s Approach*” - Roger S. Pressman. 5º Ed. (2000). McGraw Hill.
 - “*ESA Software Engineering Standards*”. PSS-05-0 Issue 2. ESA Board for Software Standardization and Control (BSSC) - European Space Agency. (1991). URL: www.ess.co.at/ECOSIM/ESA.txt
 - “*Software Engineering*” - Ian Somerville. 6º Ed. (2000). Addison Wesley.
 - “*A Discipline for Software Engineering*” – Watts Humphrey. (1995). Addison Wesley.
- Software Engineering Institute (SEI), Carnegie Mellon University (www.sei.cmu.edu)
- Web-site del curso (www.dcc.uchile.cl/~cc51a)

Guía de Supervivencia:

- Piensa, y luego actúa
- Preguntar es un síntoma de interés e inteligencia.
- Pregunta a tiempo no dejes todo para el final...
- Trata a los demás como deseas que te traten.
- Discute los problemas con sus compañeros.
- pero debes ser honesto.
- Ven a clases. Hay muchas cosas que hay que vivirlas para aprenderlas.
- Entiende que tu éxito en este curso, sólo puede ser construido por tu equipo de trabajo.
- ... Por lo tanto, sé generoso con tus conocimientos, responsable con tu trabajo, y crítico (constructivo) con el trabajo de los demás.
- Acepta la ayuda o las sugerencias de tus compañeros, ellos están tratando de mejorar el producto final.

Guía de Supervivencia:

- Asuman que van a tener problemas de comunicación y coordinación, por lo tanto adelántese a esos riesgos y generen planes de contingencia.
- Asuman que van a tener problemas de internos (por ejemplo gente enferma) y externo (con el cliente) por lo tanto adelántese a esos riesgos y generen planes de contingencia.
- Asuman que los plazos nunca son suficientes, por lo tanto adelántese a esos riesgos y generen planes de contingencia.

Tarea para el Viernes 11/3

Los que están inscritos en el curso y no lo van a tomar, por favor avísenme.

Los que no están inscritos y quieren tomar el curso, por favor avísenme.

Los que van a tomar el curso deben entregar a Magaly un currículum de no más de 3 hojas, que contenga:

- Datos Personales (incluir: teléfonos, emails y foto).
- Manejo de Inglés.
- Experiencia en desarrollo de software.
- 2 Roles que le interesaría desarrollar (1 rol preferido, y 1 rol secundario)