

# Ingeniería de Software - IIC2143

---

Yadran Eterovic S.

# Al finalizar el curso, ustedes estarán capacitados para:

---

Llevar a cabo el desarrollo de un sistema de una manera metódica

Desarrollar requisitos claros, concisos y precisos

Aplicar principios y patrones al diseñar un sistema y al evaluar el diseño de un sistema

Crear diagramas de clases en UML que modelen el dominio de un problema y la arquitectura de software de un sistema

Crear diagramas de secuencia, de estados, y de actividades en UML que modelen los casos de uso y el comportamiento de un sistema

Aplicar técnicas de testing simples a distintos niveles de un producto de software

# Contenido

---

Procesos y metodologías

Requisitos

Diseño

Arquitectura

Gestión del proyecto

Aseguramiento de calidad y *testing*

Métodos estadísticos

# El proyecto

---

A lo largo del semestre ustedes trabajarán en grupos de 3 estudiantes para desarrollar una aplicación Web

El proyecto (P) será evaluado en varias oportunidades:

- 5 entregas parciales (EP), al menos
- una entrega final (EF)
- una presentación final (PF)

# Tres interrogaciones y un examen

---

I1: 10 de sept.

I2: 11 de oct.

I3: 9 de nov.

Examen: 30 de nov.

Este curso suscribe el **Código de Honor** de la  
universidad

---

<http://www.uc.cl/codigodehonor/el-codigo>

Copias y otros serán sancionados con nota final  
***NF = 1.1*** en el curso

# ¿Cuándo *ingeniería de software*?

---

Al pasar de

... un programador, 1,000 líneas de código y dos meses de trabajo

... a

... un equipo de varios programadores, analistas, testers, etc., que escriben 1,000,000 líneas de código a lo largo de dos años de trabajo (durante los cuales cambian todas las condiciones iniciales del problema)

**El concepto cumple este año 50 años de vida**

# ¿Qué implica?

---

El problema (que se quiere resolver mediante un programa computacional) ya no lo tiene una persona (p.ej., un ingeniero militar calculando trayectorias de proyectiles)

... que es el cliente y, al mismo tiempo, el usuario (y probablemente el programador)

... sino toda una organización (p.ej., una tienda de departamentos que quiere ofrecer una nueva forma de pago a sus clientes)

... y por lo tanto el cliente y los usuarios son muchas personas distintas

... que seguramente ni siquiera están de acuerdo en qué quieren exactamente, y que van cambiando de parecer a lo largo del tiempo



# Por lo tanto, los desafíos de la disciplina

---

## A) Entender bien el problema:

- averiguar qué es lo que realmente se necesita

## B) Entender bien la solución:

- diseñar e implementar una aplicación que efectivamente resuelva el problema
- ... tomando en consideración las múltiples restricciones que provienen tanto del negocio como de la tecnología (y que posiblemente implican que la solución ideal no es factible)

## C) Planificar, coordinar, controlar el trabajo de muchas personas para pasar de A a B

# En la práctica ...

---

## **historias de terror**

Un médico, un ingeniero eléctrico y un ingeniero de software discuten sobre cuál es la profesión más antigua del mundo

Por supuesto, el médico argumenta que, como la biblia lo dice,

“... de esa costilla [la del hombre] Dios el Señor hizo una mujer ...”

claramente la profesión más antigua es la de médico cirujano

Entonces el ingeniero eléctrico interrumpe argumentando que en la misma biblia dice que, antes de la creación del hombre y la mujer,

“Dios dijo: ‘¡Que haya luz!’ Y hubo luz.”

... por lo que claramente la primera profesión fue la de ingeniero eléctrico

A lo que el ingeniero de software responde que incluso antes de la luz

“La tierra no tenía ninguna forma; todo era un mar profundo cubierto de oscuridad ...”

... o sea, reinaba el caos

... es decir, tiene que haber habido un ingeniero de software a cargo

# En realidad, a la ingeniería de software no le ha ido tan mal

---

Observemos cómo funciona el mundo moderno

# Nuestro desafío

---

De procesos y metodologías de desarrollo y de gestión de proyectos, de métodos para averiguar y documentar requisitos, de técnicas para diseñar software y para componer sistemas a partir de subsistemas, etc.

... tenemos que elegir algunas y aprenderlas en un semestre