

TEMAS

REAL. COMPUTER. SCIENCE.

TEMAS

con: Kuo Lun Lo



Liskov substitution e Interface Segregation

Introducción al principio "Liskov substitution" y "Interface Segregation" en el diseño y desarrollo del software.

Presentación de ejemplos del uso de "Liskov substitution" y "Interface Segregation", donde se muestra cómo este patrón de diseño permite reducir el acoplamiento, mejorar la reutilización del código y facilitar la realización de pruebas unitarias, generando un software de mayor calidad.

Para: Estudiantes de segundo semestre en adelante

Duración: 30 min



Single Responsibility y Open-Closed

Introducción al "Single Responsibility" y "Open-Closed" en el diseño y desarrollo del software.

Presentación de ejemplos donde se muestra cómo este patrón de diseño permite reducir el acoplamiento, mejorar la reutilización del código y facilitar la realización de pruebas unitarias, generando un software de mayor calidad.

Para: Estudiantes de segundo semestre en adelante

Duración: 30 min



Strategy Pattern

Introducción a los beneficios y el potencial en el uso de "Strategy Pattern", uno de los patrones de diseño que ayuda a entender, expresar y evitar problemas en programación.

Para: Estudiantes de segundo semestre en adelante

Duración: 30 min



TEMAS

con: *Andrés Ramírez*



Liskov substitution e Interface Segregation

Introducción al principio “Liskov substitution” y “Interface Segregation” en el diseño y desarrollo del software.

Presentación de ejemplos del uso de “Liskov substitution” y “Interface Segregation”, donde se muestra cómo este patrón de diseño permite reducir el acoplamiento, mejorar la reutilización del código y facilitar la realización de pruebas unitarias, generando un software de mayor calidad.

Para: Estudiantes de segundo semestre en adelante

Duración: 30 min



Single Responsibility y Open-Closed

Introducción al “Single Responsibility” y “Open-Closed” en el diseño y desarrollo del software.

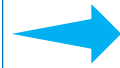
Presentación de ejemplos donde se muestra cómo este patrón de diseño permite reducir el acoplamiento, mejorar la reutilización del código y facilitar la realización de pruebas unitarias, generando un software de mayor calidad.

Para: Estudiantes de segundo semestre en adelante

Duración: 30 min

TEMAS

con: *Oscar Vásquez*



Dependency Injection

Introducción al principio “Dependency Injection” en el diseño y desarrollo del software. Presentación de ejemplos del uso de inyección de dependencias, donde se muestra cómo este patrón de diseño permite reducir el acoplamiento, mejorar la reutilización del código y facilitar la realización de pruebas unitarias, generando un software de mayor calidad.

Para: Estudiantes de segundo semestre en adelante.

Duración: 30 min.



Mi primer currículum y entrevista.

Descripción sobre buenas prácticas para la elaboración del currículum y para la preparación para la primera entrevista. Se complementa con ejemplos desde el punto de vista del entrevistador como del entrevistado.

Para: Estudiantes de último año.

Duración: 30 min.

TEMAS

con: Jorge Fallas



TFS / VSTS como herramienta para administración del ciclo de vida de las aplicaciones.

Introducción al uso de herramientas para la administración del ciclo de vida de las aplicaciones, Presentación de ejemplos de uso de estas herramientas en casos prácticos que abarcan desde el control de versiones hasta la puesta en producción de una aplicación (Sea Web, Desktop, Base de Datos).

Para: Estudiantes de segundo semestre en adelante

Duración: 30 min



TEMAS NUEVOS

con: *Juan José Mena*

- ➡ Cloud Computing
- ➡ Automatización



con: *Luis Felipe Mora*

- ➡ Test Driven Development
- ➡ Using QA Principles in Real Life
- ➡ Bases de Datos de Grafos



con: *William Méndez*

- ➡ Project Management Basics
- ➡ Agile vs Waterfall



REAL. COMPUTER. SCIENCE.

www.mobilize.net/careers-at-mobilize



GRACIAS

Carolina Madrigal

📞 4000 – 1630 / 8887 - 8905

Reclutadora

✉️ cmadrigal@mobilize.net

🌐 www.mobilize.net