

Programación Orientada a Objetos 2020

Práctica 2

Design principles y Design Patterns

- SOLID, es un acrónimo de 5 importantes principios de la programación orientada a objetos. Explicar y brindar ejemplos.
- Sobre el patrón de diseño Singleton, explique:
 - ¿Porque se puede considerar que no cumple con el principio de Single Responsibility?
 - ¿Porque se considera que dificulta realizar tests de unidad?
- Dependency Injection design pattern
 - Explique el patrón
 - ¿Cuál es la diferencia entre Dependency Inversion y Dependency Injection?
 - Brinde ejemplos de código donde se aplique.

Web Applications

- Explique los siguientes tipos de aplicaciones web
 - Dynamic Web Applications
 - Single Page Applications
 - Progressive Web Applications

Java Build Tools

- Describir:
 - Apache Ant
 - Apache Maven
 - Gradle
- Apache Maven
 - ¿Qué es el archivo POM de maven?
 - Explique brevemente el Build Lifecycle de maven y cada una de sus etapas.

- Crear un proyecto Java del tipo web application utilizando Maven.

Versionado de código

- ¿Qué son las herramientas para el versionado de código?
- ¿Cuáles son las diferencias en las herramientas de versionado de código Subversion (SVN) y GIT?
- Explique que es un workflow o branching model
- Explique en que consiste las técnica de code review / pull request
- Sobre el proyecto anteriormente creado:
 - Inicializar un repositorio Git
 - Determinar tipos de archivos o directorios que conviene no ser versionados (según el tipo de aplicación y lenguaje) e incluirlo .gitignore.
 - realizar un commit inicial con la estructura inicial del proyecto
 - crear la rama develop y sobre ella crear la clase User (con atributos email y password y sus respectivos getters y setters) en package ar.edu.unnoba.poo.2020.model.
 - realizar un nuevo commit para incluir esta modificaciones.
 - cambiar al branch master
 - realizar un merge con la rama develop
 - crear un tag especificando la version 0.1