**Primer Parcial Taller de Ingeniería de Software**

**NOMBRE: EDGAR ANDRES ROJAS APAZA**

**1.-¿Qué son los patrones de diseño?**

R.- Los patrones de diseños son formas de soluciones reutilizables a problemas comunes que surgen durante el desarrollo de software. No son código exacto, sino guías que pueden adaptarse a diferentes situaciones para mejorar la estructuración, legibilidad y mantenimiento del código.

**2. Especifique el objetivo del patrón builder**

R.- El patrón de diseño buider es pa que todo reutilizable para problemas comunes, es decir que no es código exacto si no que son guias para adaptarse a diferentes situaciones para mejorar la estructura del código y se pueda mantener a futuro.

**3. Cuales son los objetivos del clean Code**

R.- Hacer que el código sea fácil de leer y entender, facilitar el mantenimiento, reducción de errores y mejorar calidad, permitir que un código este bien estructurado y sea explicativo

**4. Describir 4 recomendaciones para evitar code smell**

R.-Evitar métodos muy largos, que sean divididas en funciones claras y no muy largas, eliminar código duplicado, para que las funciones sean reutilizables, usar nombres claros, para las clases, funciones y también para las variables y por ultimo aplicar el solid para que tenga una buena estructura el código.

**5. Indicar cuales son los principios SOLID.**

R.- S – Single Responsibility Principle (SRP): Una clase debe tener una única responsabilidad.

O – Open/Closed Principle (OCP): El código debe estar abierto para extensión pero cerrado para modificación.

L – Liskov Substitution Principle (LSP): Las clases hijas deben poder sustituir a sus padres sin romper el programa.

I – Interface Segregation Principle (ISP): No forzar a una clase a implementar interfaces que no usa.

D – Dependency Inversion Principle (DIP): Depender de abstracciones, no de implementaciones concretas.