

**Taller:**

**Refactoring**

**Integrante:**

**Victor Garcia Romero**

**Materia:**

**Diseño de software**

**2022 – 1s**

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN**

**DISEÑO DE SOFTWARE**

**Taller de Refactoring - I TÉRMINO 2020**

**Objetivos**

* Identificar malos olores de programación en el código fuente adjunto y las técnicas de refactorización correspondientes.
* Aplicar técnicas de refactorización que pueden aplicarse para eliminar los malos olores previamente identificados.

**Descripción**

En equipos de trabajo, conformados por hasta tres estudiantes, (i) identifique los malos olores de programación encontrados en el código fuente adjunto, (ii) Identifique las técnicas de refactorización adecuadas para eliminar los malos olores encontrados y (iii) Refactorice el código fuente para obtener un código más limpio y fácil de leer. Justifique su respuesta.

**Especificaciones**

Considere un **sistema académico** que permite manejar los estudiantes registrados en ciertos paralelos de distintas materias. Además, cada materia tiene un profesor y podría tener asignado un ayudante.

**Sección A**

Elabore un **reporte** en el que **identifique los code smells** encontrados en el código adjunto. Para cada code smell debe **indicar el nombre**, las **consecuencias** de mantener el mismo en dicho código y la(s) técnicas de refactorización utilizadas para eliminarlo. Para cada mal olor coloque una captura inicial y una del código refactorizado. Indique cualquier asunción que realice. **[60%]**

**Sección B**

Cree un repositorio en GitHub para subir el reporte y realizar las mejoras sobre el código inicial y aplique las técnicas indicadas en el reporte. **[40%]**

**Entregables (enlace a repositorio en Github)**

1. Un **documento** (**disponible en formatos docx o pdf**) que contenga el reporte y las imágenes correspondientes. Además, el documento debe tener una carátula en la que se liste a los integrantes del equipo, una tabla de contenido. Cada code smell debe estar en una página.
2. El repositorio debe contener en el directorio raíz el documento indicado anteriormente y la carpeta del proyecto refactorizado. Subir a Sidweb el enlace a dicho repositorio.
3. Cada estudiante debe entregar una evaluación a los miembros de su equipo de trabajo usando la rúbrica de co-evaluación en la **tarea individual** “Co-evaluación Taller de Refactoring” a disposición en SidWeb.

**Rúbrica de Calificación**

|  |  |
| --- | --- |
| **Sección A** | **Valor** |
| Identificar al menos 6 malos olores | 30 |
| Identificar técnicas para eliminar dichos malos olores. | 30 |
| **Sección B** |  |
| Aplicar correctamente las técnicas de refactorización indicadas en el reporte | 40 |
| **Penalidades** | **Valor** |
| Penalidad por hora o fracción de hora de retraso en la entrega | -30 |
| No subir a SidWeb los entregables requeridos de acuerdo con lo especificado | -50 |
| No subir la rúbrica de coevaluación en tarea individual (afectación individual) | -100 |

## Code Smells

Texto

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

figura1: Inappropriate intimacy figura2: Large Class & Divergente Change

Texto

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura3: Temporary Field Figura4: Temporary Field

Imagen que contiene Texto

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura5: Inappropriate intimacy figura6: Inappropriate intimacy

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente Texto

Descripción generada automáticamente

Figura7: Shotgun surgery Figura8: Dead code

## Refactoring Techniques

Para *Inappropriate intimacy (figura1,5 y6)* utilize *Hide Delegate.*

Para *figura2* Change utilize *Replace Delegation with Inheritance.*

Para *Temporary Field* (figura 3y4) utilize *Replace Method with Method Object.*

Para figura 7 y figura 8 se uso *Move Method* y *Inline Class.*