|  |  |
| --- | --- |
| **Instituto Tecnológico de Costa Rica**  **Ingeniería en Computación**  **Campus San Carlos** | Primer Proyecto Programado  Programación Orientada a Objetos |
| Prof. Oscar Víquez Acuña. | Sistema “AllCanBeFixed” |

**Objetivos**

* Aprender el lenguaje Java y familiarizarse con la Programación Orientada a Objetos.
* Aplicar conceptos de diseño de aplicaciones utilizando la teoría de Orientación a Objetos.
* Generar una herramienta que canalice todos los conceptos estudiados del tema.

**Descripción del proyecto**

El proyecto consiste en implementar una simulación del funcionamiento de una tienda de reparación y venta de equipos electrónicos. Inicialmente se debe diseñar la estructura de clases en Java que represente los elementos esenciales básicos modelables en dicho sistema; para lo cual se debe conocer/investigar el funcionamiento hipotético de este tipo de negocios.

La tienda “All can be fixed” repara artículos electrónicos y vende artículos nuevos. Pertenece a un sistema de tiendas a nivel nacional. Algunas de sus características son:

* Repara y vende solamente artículos electrónicos de diversa índole.
* El sistema de cómputo de la tienda debe llevar el control del inventario de artículos reparados, artículos no reparados y artículos vendidos. Para esto y con fines meramente académicos, se desea que se utilice **una única estructura de datos** afín para cada uno de los artículos mencionados (aunque potencialmente tengan diferentes elementos).
* La información pertinente a cada artículo queda a decisión del programador pero se debe considerar que hay información común e información diferente según el tipo de artículo para aprovechar las bondades que da la herencia en este tipo de paradigmas de programación.
* El sistema también debe tener un módulo para realizar facturación tanto de los artículos reparados como de los artículos vendidos. Las facturas serán un poco diferentes debido a ciertos campos extra como por ejemplo reparado o no reparado y motivo de dicho estado del artículo. Se debe manejar un historial de ventas y reparaciones por fecha y montos acumulados por semana y por mes. De la misma forma con los artículos, el contenido de cada factura se deja a criterio del estudiante siempre que sea acorde con lo que el problema amerita. Se debe implementar una estructura similar a la de artículos de manera que sea genérica y única para almacenar todas las facturas.
* El sistema además tendrá un módulo de planillas donde se manejará la planilla total de la empresa. Cada empleado tiene rango diferente dentro de la tienda y escoge la forma en que quiere que la empresa le pague, por lo que estos datos deben considerarse al momento de elegir las características de cada clase de empleado. Se debe separar las abtracciones que consideren los datos de la persona junto con los datos propios del empleado. Este módulo se encarga de calcular las planillas semanales, quincenales y mensuales, así como un historial de pagos. De igual forma, este módulo debe ser genérico y se debe utilizar una sola estructura de datos.
* El dueño del conjunto de tiendas quiere que este sistema le funcione en las demás tiendas sin necesidad de hacer muchos cambios por lo que debe pensarse en que la especificación que se realice sea lo más reutilizable posible. Las estructuras de datos y los métodos deben ser bastante modulares y genéricas para permitir dicha característica. SE RECOMIENDA QUE SE UTILICEN INTERFACES O CLASES ABTRACTAS PARA MODELAR TIPOS GENÉRICOS.

Se pide que el estudiante diseñe el modelo de clases utilizando al máximo los conceptos de Orientación a Objetos y que implemente en Java las clases necesarias, según se especificó en el modelo creado. Para probar las clases se solicita que hagan Interfaz Gráfica de Usuario (GUI por sus siglas en inglés), de manera que para probar el funcionamiento de los módulos, se debe poder realizar en dicha interfaz todas las operaciones que el problema plantea en términos de reparaciones y ventas de artículos así como reportes de facturación y planillas.

No se solicita puntualmente que la interfaz tenga la capacidad de administrar la información base (artículos, facturas y empleados) por lo que el sistema puede iniciar a partir de un conjunto de información “quemada” previamente, que sirva como base para probar las demás operaciones mencionadas.

Para el diseño del modelo de clases se debe utilizar UML estándar sin excepción. Se debe realizar el modelo de clases (características y métodos de funcionamiento) y las relaciones entre cada clase.

**Documentación:**

El desarrollo del proyecto debe documentarse adecuadamente en varias etapas del mismo. A saber se enumeran los documentos que deben adjuntarse al finalizar el tiempo del proyecto junto con el código fuente del programa:

* Diagrama de Clases: El diagrama de clases **debe entregarse de manera definitiva exactamente 15 días naturales posteriores a la fecha de inicio del proyecto**. Este diagrama puede ser validado por el profesor/asistente/tutor antes de la fecha de entrega para garantizar un proceso mayor de análisis y debate en torno a la propuesta diseñada.
* Documentación formal: La documentación formal estará compuesta por los siguientes apartados:
  + Análisis del Problema: En donde se analiza claramente el problema desde la perspectiva del estudiante y las implicaciones que este tiene. Es deseable establecer en este apartado, un objetivo general y objetivos específicos que al finalizar el proyecto y la documentación son contrastados con los resultados obtenidos.
  + Análisis de la Solución planteada: Donde se muestran y analizan los pormenores relacionados con el diseño del proyecto, las tecnologías utilizadas y los algoritmos planteados (documentando adecuadamente aquellos que no son propiamente de su autoría)
  + Resultados Obtenidos: Donde se contrasta los resultados obtenidos contra los objetivos del proyecto haciendo énfasis en los detalles complejos de implementar y los aspectos que pudieron eventualmente no haber sido concluidos.
  + Conclusiones y recomendaciones: sobre elementos propios del desarrollo del software
* Manual de Usuario: Un breve video subido a la plataforma Youtube de forma pública que explique los pormenores referentes al uso del sistema por parte de sus eventuales usuarios. Se debe anotar el link del video.

**Notas Finales para la Tarea:**

* La tarea es en parejas y debe realizarse en cualquier editor de Java. Bajo ninguna circunstancia se permitirán copias de trabajos, caso contrario se hará acreedor de las sanciones que establece el reglamento.
* Fecha de entrega para el proyecto: Lunes 15 de Mayo de 2023 antes de las 10:00 pm. La entrega debe hacerse a través del TEC-DIGITAL.