Enunciado Proyecto 1

## OBJETIVOS:

* Aplicar los conocimientos adquiridos durante el curso para el desarrollo de una aplicación de software basada en las mejores prácticas vistas en clase.
* Codificar soluciones para un problema dado por medio de un enfoque modular.
* Creación y manipulación de distintas estructuras de datos en C#.

## Descripción del Proyecto:

El objetivo de este proyecto es que los estudiantes apliquen sus conocimientos de estructuras de datos en la implementación de un sistema de gestión de tarjetas de crédito en memoria. Utilizando el lenguaje de programación C#, los estudiantes deberán desarrollar un conjunto de APIs REST que permitan realizar operaciones básicas relacionadas con tarjetas de crédito, tales como consultas de saldo, pagos, y generación de estados de cuenta. Toda la información manejada por estas APIs deberá almacenarse en memoria utilizando estructuras de datos como pilas, colas, listas enlazadas y árboles binarios de búsqueda. El sistema deberá cargar inicialmente la información de las tarjetas de crédito desde un archivo en formato JSON.

## Requisitos Funcionales:

1. API de Carga Inicial: Leer y cargar los datos de las tarjetas de crédito desde un archivo JSON a las estructuras de datos en memoria.
2. API de Consulta de Saldo: Utilizar listas enlazadas para almacenar y consultar el saldo de las tarjetas.
3. API de Realización de Pagos: Emplear colas para gestionar los pagos realizados a las tarjetas de crédito, actualizando el saldo correspondiente.
4. API de Generación de Estados de Cuenta: Utilizar árboles binarios de búsqueda para organizar y generar los estados de cuenta de cada tarjeta.
5. API de Consulta de Movimientos: Implementar pilas para almacenar y consultar los últimos movimientos realizados con la tarjeta.
6. API de Notificaciones: Implementar una cola para gestionar las notificaciones en tiempo real que se enviarán a los usuarios, como alertas de transacciones y recordatorios de pago.
7. API de Cambio de PIN: Utilizar una lista enlazada para almacenar y gestionar los cambios de PIN de las tarjetas de crédito.
8. API de Bloqueo Temporal: Emplear un árbol binario de búsqueda para organizar y gestionar el estado de bloqueo de las tarjetas de crédito.
9. API de Solicitud de Aumento de Límite de Crédito: Utilizar una pila para almacenar y procesar las solicitudes de aumento de límite de crédito.

## Requisitos No Funcionales:

1. Eficiencia en el Manejo de Memoria: El sistema debe optimizar el uso de memoria al almacenar y gestionar los datos en las estructuras de datos.
2. Rendimiento: El sistema debe ser capaz de realizar las operaciones de manera rápida y eficiente, aprovechando las características de las estructuras de datos utilizadas.
3. Seguridad: Asegurar la protección de los datos almacenados en memoria, implementando mecanismos para prevenir accesos no autorizados.
4. Escalabilidad: El diseño del sistema debe permitir la fácil incorporación de nuevas funcionalidades o estructuras de datos en el futuro.

## Entregables:

* Código Fuente: Implementación en C# de las APIs REST y las estructuras de datos en memoria.
* Archivo JSON de Datos Iniciales: Archivo con los datos de las tarjetas de crédito que se utilizarán para la carga inicial del sistema.
* Documentación de APIs: Descripción de los endpoints de las APIs, incluyendo parámetros, formatos de solicitud y respuesta.
* Informe Técnico: Documento que explique el diseño e implementación de las estructuras de datos y cómo estas se integran en el sistema.

## Entregables:

|  |
| --- |
| **ENTREGABLE** |
| 1. Un proyecto que genere la plataforma de desarrollo que está usando. |
| 1. Código fuente en un repositorio de Github |
| 1. El desarrollo debe realizarse en lenguaje C# y utilizando la herramienta de desarrollo como Visual Studio |
| 1. Un archivo readme.txt para indicar como ejecutar el programa. |
| 1. Un video tutorial indicando el funcionamiento de la aplicación. |

## Se evaluará:

* Adecuada aplicación de los conocimientos.
* Calidad de la programación: Identificación correcta de los nombres de las clases, propiedades, variables y métodos.

## Asuntos importantes:

* Este proyecto se deberá realizar en grupos de 2 personas, de no cumplir con esta regla quedará sin nota automáticamente.
* Los proyectos pasarán por un proceso de investigación, que permitirá demostrar las copias que puedan existir, por lo que, de encontrarlos automáticamente todas las personas involucradas tendrán 0 en su nota y sin derecho a presentar el siguiente proyecto.
* Para este proyecto la ponderación es de 10 puntos.