UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA

**FACULTAD DE INGENIERÍA EN INFORMÁTICA Y SISTEMAS**



**Proyecto de Programación Básica**

**TÍTULO DEL PROYECTO:**

**Proyecto sobre código de monedero virtual.**

**CURSO:**

**Programación Básica.**

**DOCENTE:**

**Christian García Villegas.**

**INTEGRANTES:**

**Carhuancho Osorio, Alesander Enzo.**

**Lopez Vasquez, Davis Franchesco.**

**Ventura Canales, Hiyno Agustin.**

**FECHA DE ENTREGA: *13/03/2025***

**SEMESTRE:**

**2025-0**

**ÍNDICE**

[**Cuadro de versiones :** 1](#_Toc193143869)

[INTRODUCCIÓN 2](#_Toc193143870)

[**Problema :** 3](#_Toc193143871)

[**Objetivos** 5](#_Toc193143872)

[**Posibles Interesados y/o Personas a Quienes va Dirigido el Código:** 6](#_Toc193143873)

[**Tecnologías y Herramientas Utilizadas:** 7](#_Toc193143874)

[**Justificación:** 7](#_Toc193143875)

[DISEÑO DEL SISTEMA 8](#_Toc193143876)

[ **Requisitos del sistema:** 8](#_Toc193143877)

[ **Funcionales** 8](#_Toc193143878)

[ **No funcionales** 8](#_Toc193143879)

[ **Análisis de Requerimientos:** 8](#_Toc193143880)

[ **Entradas** 8](#_Toc193143881)

[ **Salidas** 8](#_Toc193143882)

[ **Restricciones** 8](#_Toc193143883)

[**Diagrama de Clases:** 9](#_Toc193143884)

[**Descripción de las Clases:** 10](#_Toc193143885)

[IMPLEMENTACIÓN: 11](#_Toc193143886)

[**Fragmentos del Código fuente:** 11](#_Toc193143887)

[**Ejecución del programa:** 18](#_Toc193143888)

[PRUEBAS: 19](#_Toc193143889)

[**Casos de prueba y Validación del sistema:** 19](#_Toc193143890)

[CONCLUSIONES 21](#_Toc193143891)

[**Anexos** 22](#_Toc193143892)

[**Código completo** 22](#_Toc193143893)

[**Diagrama** 37](#_Toc193143894)

[**URL Git** 38](#_Toc193143895)

[**Video en YouTube** 38](#_Toc193143896)

**Cuadro de versiones:**

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Elaborado por: | Revisado por: | Fecha |
| 0.5 | Todo el grupo | **Christian García Villegas** | 12/02/2025 |
| 0.12 | Todo el grupo | Todo el grupo | 21/02/2025 |
| 0.15 | Todo el grupo | **Christian García Villegas** | 26/02/2025 |
| 0.20 | Todo el grupo | Todo el grupo | 03/03/2025 |
| 0.30 | Todo el grupo | **Christian García Villegas**  (Por ser revisado) | 13/03/2025 |

INTRODUCCIÓN

La educación financiera es un aspecto fundamental para el desarrollo personal y profesional, especialmente en estudiantes y jóvenes adultos. Sin embargo, muchas personas no cuentan con los conocimientos necesarios para administrar correctamente su dinero, lo que puede llevar a problemas económicos a lo largo de su vida. La falta de planificación y control sobre los ingresos y gastos puede resultar en endeudamiento excesivo, falta de ahorro y dificultades para alcanzar metas financieras a futuro.

Este proyecto tiene como objetivo desarrollar un monedero virtual programado en C++, diseñado específicamente para ayudar a estudiantes a gestionar mejor sus finanzas personales. A través de una plataforma sencilla e intuitiva, los usuarios podrán registrar ingresos y gastos, planificar su presupuesto y obtener recomendaciones para mejorar su administración financiera. La herramienta permitirá a los estudiantes visualizar de manera clara sus movimientos financieros, analizar sus patrones de gasto y tomar decisiones más informadas sobre su dinero.

El monedero virtual se enfoca en fomentar hábitos financieros responsables y en brindar herramientas educativas que permitan a los estudiantes aprender sobre el ahorro, el control de gastos y la toma de decisiones económicas inteligentes. Además, se busca generar conciencia sobre la importancia de administrar adecuadamente los recursos financieros desde una edad temprana, promoviendo la autonomía y la estabilidad económica a largo plazo.

Con el desarrollo de esta aplicación, también se pretende incentivar la digitalización de las finanzas personales, aprovechando el uso de tecnología para optimizar la gestión del dinero. Se espera que este monedero virtual se convierta en una herramienta clave para los estudiantes, proporcionándoles un medio accesible y seguro para mejorar su capacidad de planificación y toma de decisiones financieras.

**Problema :**

El manejo inadecuado del dinero es un problema recurrente entre estudiantes y personas jóvenes, lo que puede generar dificultades económicas a corto y largo plazo. La falta de educación financiera desde temprana edad provoca que muchos jóvenes enfrenten retos al administrar su presupuesto, lo que puede derivar en problemas como deudas innecesarias, escasez de ahorros y una dependencia excesiva de créditos o préstamos. Algunos de los problemas principales incluyen:

1. **Falta de Conocimiento Sobre Administración Financiera**
   * Muchas personas no saben cómo gestionar un presupuesto, planificar sus gastos o priorizar sus necesidades económicas. Sin una guía adecuada, el dinero suele destinarse a gastos impulsivos o innecesarios sin considerar el ahorro o la inversión en necesidades futuras.
2. **Gastos Innecesarios y Falta de Ahorro**
   * La ausencia de hábitos financieros saludables lleva a un consumo desmedido sin previsión para el futuro. La facilidad de acceso a productos y servicios digitales ha fomentado un comportamiento de gasto compulsivo, lo que dificulta la capacidad de los estudiantes para administrar su dinero de manera eficiente.
3. **Dificultad Para Realizar Un Seguimiento de Ingresos y Egresos**
   * Sin una herramienta adecuada, muchas personas pierden el control de sus finanzas, lo que genera un desbalance entre los ingresos y los gastos. La falta de registro financiero impide conocer en qué se está gastando el dinero, dificultando la toma de decisiones acertadas para mejorar la administración económica personal.
4. **Endeudamiento Temprano**
   * El mal manejo del dinero puede llevar a la adquisición de deudas innecesarias, afectando la estabilidad financiera en el futuro. Los créditos estudiantiles, las tarjetas de crédito y los préstamos personales son herramientas que pueden convertirse en un problema si no se utilizan con responsabilidad y sin un plan de pago estructurado.
5. **Falta de Herramientas Accesibles Para la Educación Financiera**
   * Existen pocas plataformas diseñadas específicamente para ayudar a estudiantes a mejorar su administración del dinero. La mayoría de las aplicaciones de gestión financiera están orientadas a personas con ingresos estables, sin considerar las necesidades específicas de los estudiantes, quienes suelen manejar ingresos variables y gastos específicos relacionados con su formación académica.

El desarrollo de un monedero virtual educativo busca mitigar estos problemas al proporcionar a los estudiantes una herramienta accesible, fácil de usar y diseñada específicamente para mejorar su capacidad de administración financiera. Con funcionalidades como registro de ingresos y gastos, generación de informes y consejos financieros personalizados, se espera fomentar una cultura de educación financiera que beneficie a los estudiantes en su vida diaria y en su futuro económico.

**Objetivos**

El desarrollo de este código tiene como propósito proporcionar una herramienta accesible y eficiente que ayude a los estudiantes a mejorar su administración financiera personal. Se enfoca en brindar una solución tecnológica que no solo sirva como una aplicación para el manejo del dinero, sino que también eduque a los usuarios en conceptos financieros esenciales. Los objetivos específicos incluyen:

* **Fomentar La Educación Financiera**
  + Proveer una guía práctica que enseñe sobre ahorro, control de gastos y planificación financiera mediante ejemplos funcionales en el código.
* **Desarrollar Un Código Accesible y Fácil de Entender**
  + Diseñar una estructura clara y bien documentada, permitiendo que cualquier persona con conocimientos básicos en programación pueda utilizarlo y adaptarlo a sus necesidades.
* **Facilitar El Seguimiento de Ingresos y Egresos**
  + Incluir funcionalidades para registrar y visualizar transacciones, ofreciendo un panorama claro de la situación financiera personal.
* **Promover El Ahorro y La Toma de Decisiones Responsables**
  + Incluir métodos que ayuden a los usuarios a establecer metas de ahorro y a evaluar sus patrones de gasto.
* **Ofrecer Recomendaciones y Tips de Gestión Financiera**
  + Integrar funciones que sugieran buenas prácticas para optimizar el uso del dinero, ayudando a evitar gastos innecesarios y promoviendo hábitos financieros saludables.

**Posibles Interesados y/o Personas a Quienes va Dirigido el Código:**

Este código está diseñado para beneficiar a distintos grupos de usuarios, ofreciendo una herramienta que no solo ayuda a administrar el dinero, sino que también educa en buenas prácticas financieras. Los principales beneficiarios incluyen:

* **Estudiantes de Nivel Secundario y Universitario**
  + Jóvenes que buscan mejorar su administración financiera personal y aprender sobre la importancia del ahorro y la gestión de gastos a través de ejemplos de código prácticos y aplicables.
* **Docentes y Centros Educativos**
  + Profesores y escuelas que desean integrar herramientas tecnológicas en sus programas de educación financiera, facilitando a los estudiantes la comprensión de conceptos financieros a través de implementaciones en C++.
* **Padres y tutores**
  + Personas interesadas en enseñar a sus hijos o familiares jóvenes sobre el manejo adecuado del dinero mediante un enfoque práctico y aplicable.
* **Emprendedores y pequeños comerciantes**
  + Individuos que buscan una solución básica y accesible para gestionar sus ingresos y egresos personales de manera estructurada y sin necesidad de software complejo.
* **Organizaciones que promueven la inclusión financiera**
  + ONGs y entidades gubernamentales que trabajan en la educación financiera y en la provisión de herramientas para la gestión del dinero en poblaciones con menos acceso a servicios bancarios.

Con esta herramienta, se espera impactar positivamente en la formación financiera de los usuarios, permitiendo una mejor planificación y administración de los recursos económicos desde una edad temprana, fomentando hábitos de ahorro y control de gastos de manera accesible y práctica.

**Tecnologías y Herramientas Utilizadas:**

En este proyecto se ha utilizado el lenguaje de programación **C++**, un lenguaje robusto y de alto rendimiento que permite la implementación de paradigmas avanzados, como la Programación Orientada a Objetos (POO). Se han empleado conceptos modernos de ingeniería de software (herencia múltiple, agregación, composición y asociación) que garantizan la escalabilidad y mantenibilidad del sistema. Asimismo, se han realizado diseños mediante **diagramas UML** para visualizar de forma gráfica la estructura del sistema y las relaciones entre las distintas clases. Todo ello, acompañado de un ambiente de desarrollo que facilita la compilación, depuración y pruebas del código, asegurando que cada módulo cumpla con los estándares de calidad exigidos. La elección de estas tecnologías y herramientas responde a la necesidad de crear una solución eficiente y fácil de entender para todos los integrantes del equipo.

**Justificación:**

El proyecto adquiere una relevancia especial en el contexto actual, en el que la educación financiera se posiciona como una herramienta fundamental para el desarrollo personal y profesional. El monedero virtual no solo permite a los usuarios gestionar de manera óptima sus ingresos y gastos, sino que también actúa como una plataforma educativa que fomenta el ahorro, la planificación y la toma de decisiones responsables. La implementación en C++ y el uso de la Programación Orientada a Objetos permiten estructurar el código de forma modular y escalable, lo que facilita futuras ampliaciones y mejoras. Además, la utilización de diagramas UML y otros recursos visuales contribuye a una mejor comprensión del sistema, tanto para desarrolladores como para usuarios finales. En resumen, este proyecto no solo es una solución tecnológica para la administración de finanzas personales, sino también una experiencia de aprendizaje que refuerza los conocimientos adquiridos en el ámbito de la programación y el diseño de sistemas.

DISEÑO DEL SISTEMA

* **Requisitos del sistema:**

### **Funcionales**

* + Registro de usuarios según su tipo (Estudiante, Docente, Padre/Tutor, Emprendedor).
  + Autenticación y eliminación de usuarios (con doble confirmación).
  + Registro y modificación de ingresos y gastos.
  + Visualización de un resumen financiero en forma de tabla de porcentajes.
  + Visualización de diccionario financiero y tips del día.

### **No funcionales**

* + Validación de datos (por ejemplo, fechas correctas, montos positivos).
  + Seguridad en el acceso (uso de contraseñas).
  + Persistencia de datos en archivos (opcional en futuras extensiones).
* **Análisis de Requerimientos:**

### **Entradas**

* Nombre y contraseña del usuario.
* Tipo de usuario (estudiante, profesor, padres y/o apoderados, emprendedores).
* Fuente de ingresos (trabajo o mesada) y frecuencia (diaria, semanal, mensual).
* Registro de ingresos con descripción y monto.
* Registro de gastos con categoría y monto.
* **Salidas**
* Confirmación de usuario registrado o autenticado.
* Lista de ingresos y egresos registrados.
* Resumen financiero con balance total.
* Tabla con ingresos y egresos detallados.
* Barra de distribución financiera (ahorro, gastos, inversión).
* **Restricciones**
* Máximo cinco usuarios registrados.
* Validación de usuario mediante contraseña.
* Los gastos no pueden superar los ingresos.
* Los ingresos deben ser valores numéricos positivos.

**Diagrama de Clases:**

El diagrama UML refleja las siguientes relaciones:

* **Herencia:** La clase abstracta *Usuario* es extendida por *UsuarioEstudiante*, *UsuarioDocente*, *UsuarioPadreTutor* y *UsuarioEmprendedor*.
* **Agregación:** La clase *SistemaUsuarios* gestiona un conjunto de objetos *Usuario*.
* **Composición:** Las clases *DiccionarioFinanciero* y *ConsejosFinancieros* son utilizadas para mostrar información complementaria.
* Texto

  Descripción generada automáticamente**Asociación:** Se establece entre las clases *Usuario* y *Fecha* para registrar la fecha de creación del usuario.

Texto

Descripción generada automáticamente

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Descripción de las Clases:**

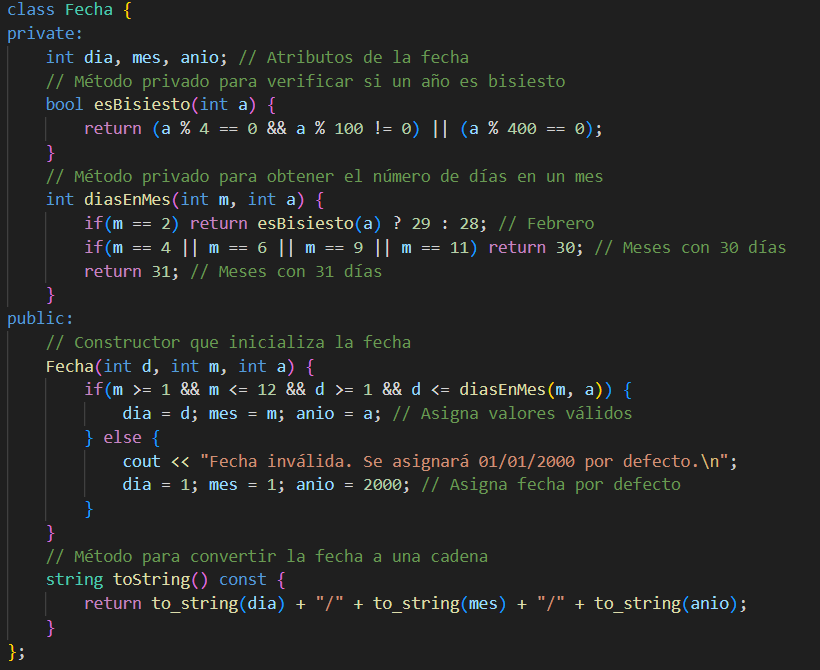
* **Fecha:**
  + **Atributos:** día, mes, año.
  + **Métodos:** Constructor (valida la fecha usando métodos para año bisiesto y días exactos por mes) y toString() para retornar la fecha formateada.
  + **Relación:** Asociada a cada usuario para indicar la fecha de creación.
* **Usuario (abstracta):**
  + **Atributos:** nombre, contraseña, edad, fechaCreacion, ingresos, gastosPrimarios, gastosSecundarios, gastosTerciarios, ahorro, inversion, estadoFamiliar.
  + **Métodos:** Métodos virtuales para ingresar datos, mostrar información y generar la tabla de gastos; método de autenticación.
  + **Relación:** Clase base para las clases derivadas.
* **UsuarioEstudiante:**
  + **Atributos específicos:** recibeMesada, mesada, frecuenciaMesada, tieneTrabajo, ingresoTrabajo, gastaEnPasaje, tieneMovilidad, loLlevan, viveSolo, alquiler, gastoComida, gastosSecundariosEst, gastosTerciariosEst.
  + **Métodos:** Implementa el método de ingreso de datos específico para estudiantes, muestra la información y genera una tabla comparativa según porcentajes recomendados (por ejemplo, 50%-20%-10%-10% para estudiantes solteros).
* **UsuarioDocente, UsuarioPadreTutor, UsuarioEmprendedor:**
  + Cada clase extiende *Usuario* y añade atributos específicos según el perfil (por ejemplo, materia en docentes, gastos familiares en padres/tutores, y datos de negocio en emprendedores).
  + Implementan métodos para ingresar datos particulares y generar tablas de resumen acorde a sus perfiles.
* **SistemaUsuarios:**
  + **Atributos:** Vector de punteros a *Usuario*.
  + **Métodos:** Agregar usuario, autenticar usuario y eliminar usuario (con doble confirmación y reingreso de contraseña).
* **DiccionarioFinanciero y ConsejosFinancieros:**
  + Permiten mostrar definiciones y consejos financieros que apoyan la educación financiera del usuario.

IMPLEMENTACIÓN:

A continuación, se muestran fragmentos de algunas de las clases más importantes (el código completo se encuentra en el anexo):

**Fragmentos del Código fuente:**

**Clase Fecha**

****

**Clase Usuario (abstracta)**

****

**Clase UsuarioEstudiante**

****

**Clase Usuario PadreTutor**

**Clase UsuarioEmprendedor**

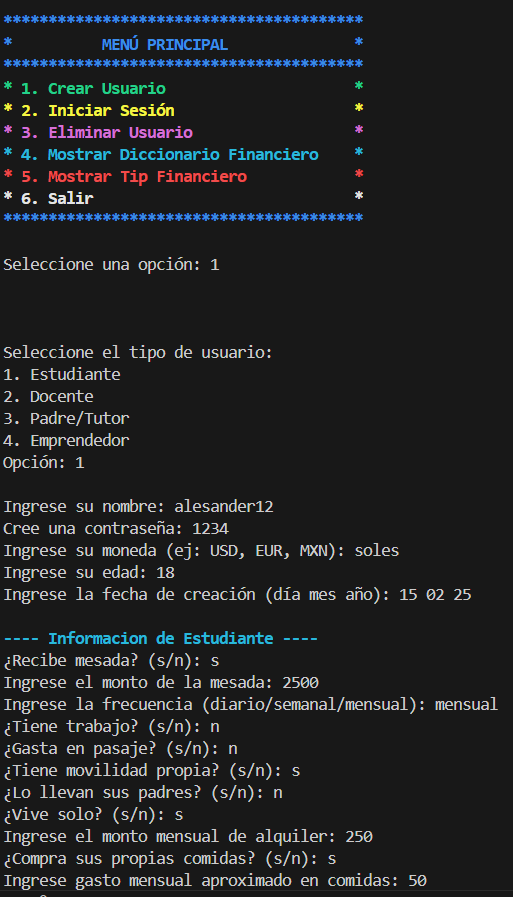
****

**Clase UsuarioDocente**

****

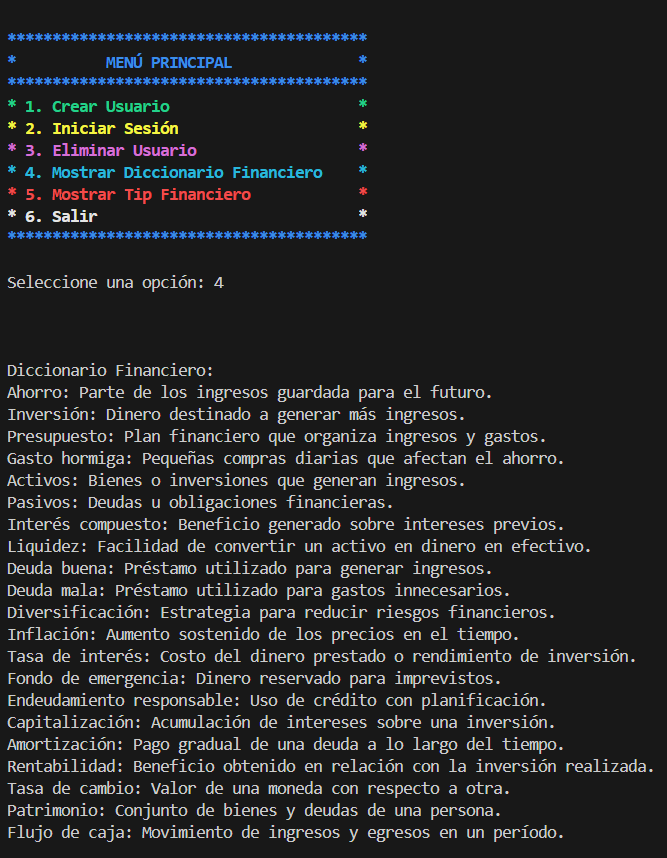
**Clase SistemaUsuarios**

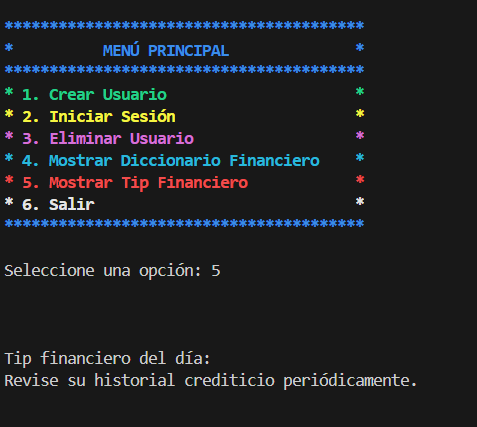
****

**Ejecución del programa:**

**Texto

Descripción generada automáticamente**

****

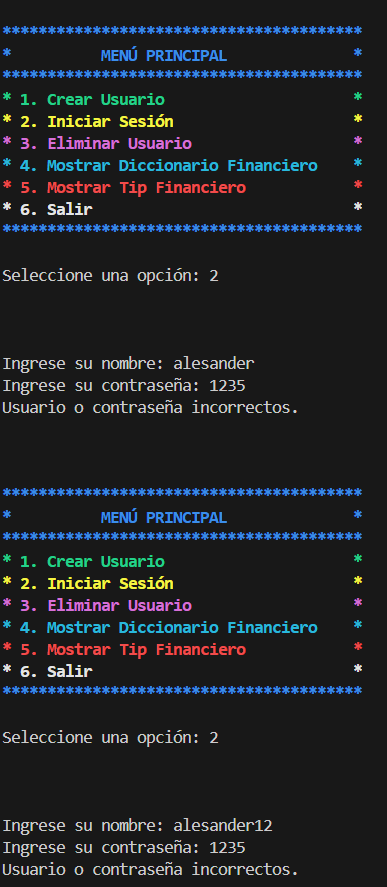
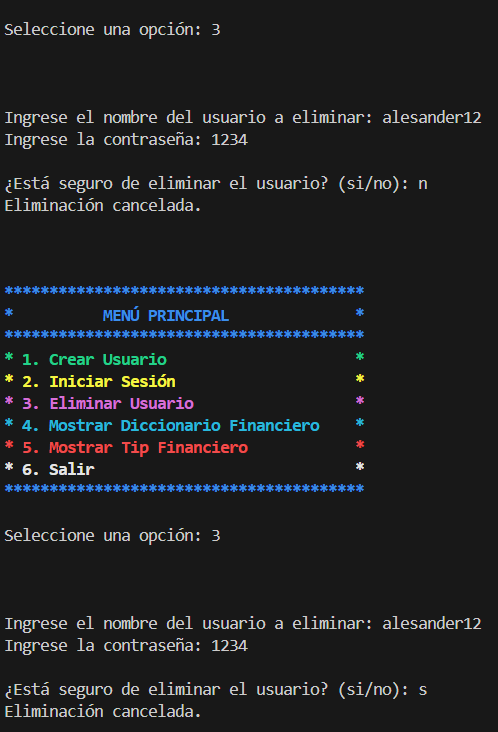
****

PRUEBAS:

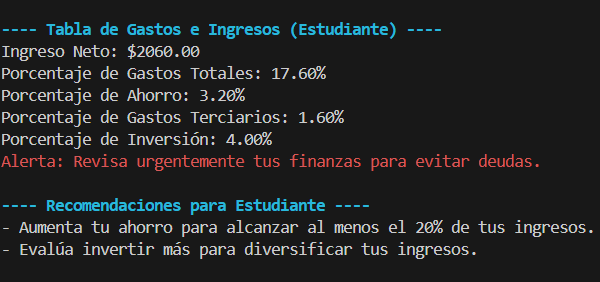
**Casos de prueba y Validación del sistema:**

Se llevaron a cabo diversas pruebas para verificar el correcto funcionamiento del sistema:

* **Pruebas de Creación de Usuarios:**  
  Se verificó que, para cada tipo de usuario (Estudiante, Docente, Padre/Tutor, Emprendedor), los datos se ingresen correctamente y se almacenen de acuerdo con las especificaciones. Se realizaron pruebas con diferentes fuentes de ingresos (trabajo o mesada) y se validó el cálculo de porcentajes en la tabla financiera.
* **Pruebas de Autenticación y Eliminación:**  
  Se comprobó que el sistema permite iniciar sesión solo con las credenciales correctas y que el proceso de eliminación de usuarios, con doble confirmación y reingreso de contraseña, se ejecute de forma segura y precisa.



* **Pruebas de Cálculo Financiero:**  
  Se realizaron escenarios con datos reales y simulados para evaluar que los porcentajes de gastos, ahorro e inversión se calculen conforme a lo esperado (por ejemplo, comparando los resultados con los valores recomendados: 50%-20%-10%-10% para estudiantes solteros). Además, se verificó la generación de mensajes de alerta (en “colores” mediante códigos ANSI) en función de las desviaciones detectadas.



* **Pruebas de Interfaz:**  
  Se simuló la interacción del usuario con el menú principal, comprobando la correcta navegación entre las opciones de creación, autenticación y eliminación, así como la visualización de información financiera detallada.

Los resultados obtenidos fueron satisfactorios, evidenciando que el sistema responde adecuadamente a entradas válidas e inválidas, y cumpliendo con todas las restricciones establecidas.



CONCLUSIONES

* **Logros obtenidos:**  
  Se desarrolló un sistema modular y escalable que integra múltiples conceptos de POO (herencia, agregación, composición y asociación) para gestionar finanzas personales de manera eficiente.
* **Dificultades encontradas:**  
  La validación de fechas y el manejo de entradas numéricas representaron desafíos importantes, los cuales se superaron mediante funciones específicas y pruebas exhaustivas.
* **Aprendizajes adquiridos:**  
  Se reforzó la importancia de un diseño modular y bien documentado, que facilita tanto el mantenimiento del sistema como futuras ampliaciones. Además, se comprobó que el conocimiento adquirido en clase es suficiente para alcanzar metas ambiciosas.
* **Posibles mejoras:**  
  Integrar una interfaz gráfica, persistencia de datos en archivos o bases de datos, y ampliar la funcionalidad para incluir nuevos módulos (por ejemplo, gestión de deudas o inversiones más avanzadas).

El desarrollo de este código ha permitido la creación de una herramienta eficiente y accesible para la administración financiera personal, adaptada a diferentes tipos de usuarios como estudiantes, profesores, padres/apoderados y emprendedores. A través de un enfoque basado en la Programación Orientada a Objetos, se logró una estructura modular que facilita la gestión de usuarios, el registro de ingresos y gastos, y la generación de reportes financieros detallados.

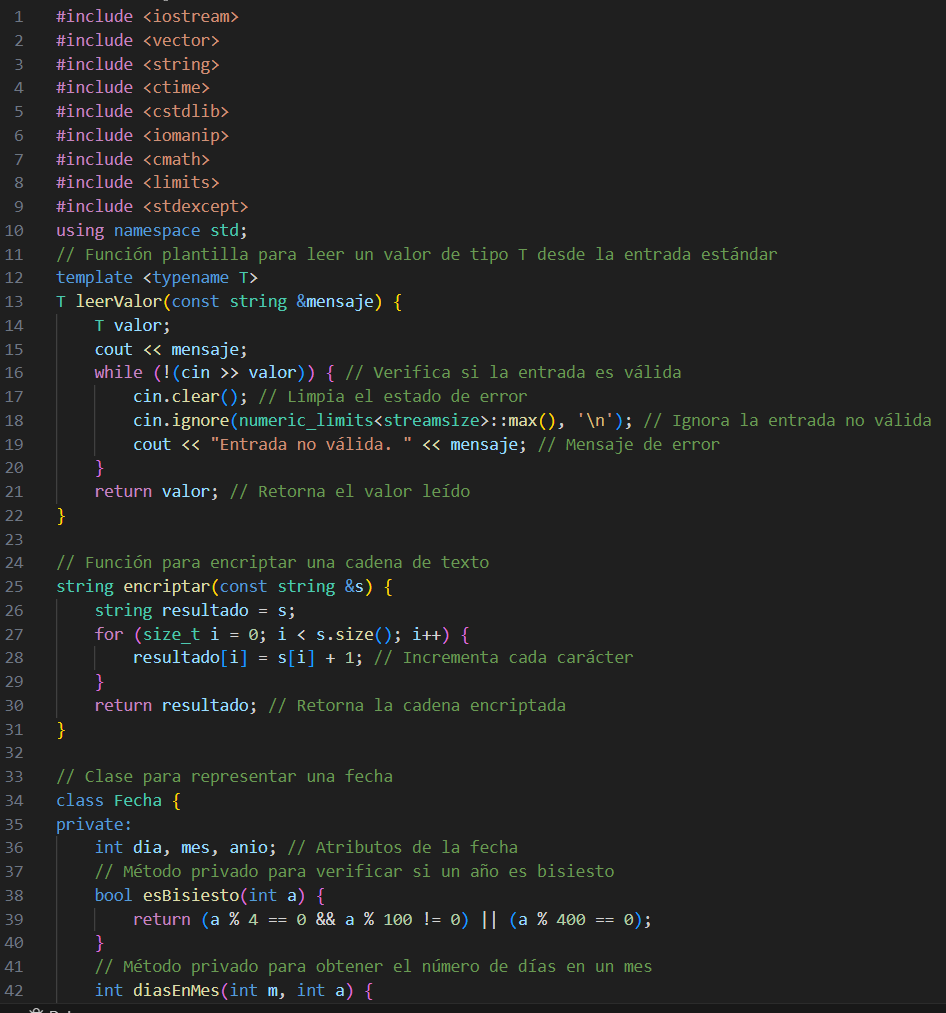
Uno de los aspectos más relevantes de este código es la incorporación de medidas de seguridad, como la autenticación por contraseña y la limitación en el número de usuarios, asegurando un uso eficiente y controlado de la aplicación. Además, la clasificación de ingresos y gastos proporciona un mayor nivel de detalle, permitiendo a los usuarios evaluar su situación financiera con precisión.

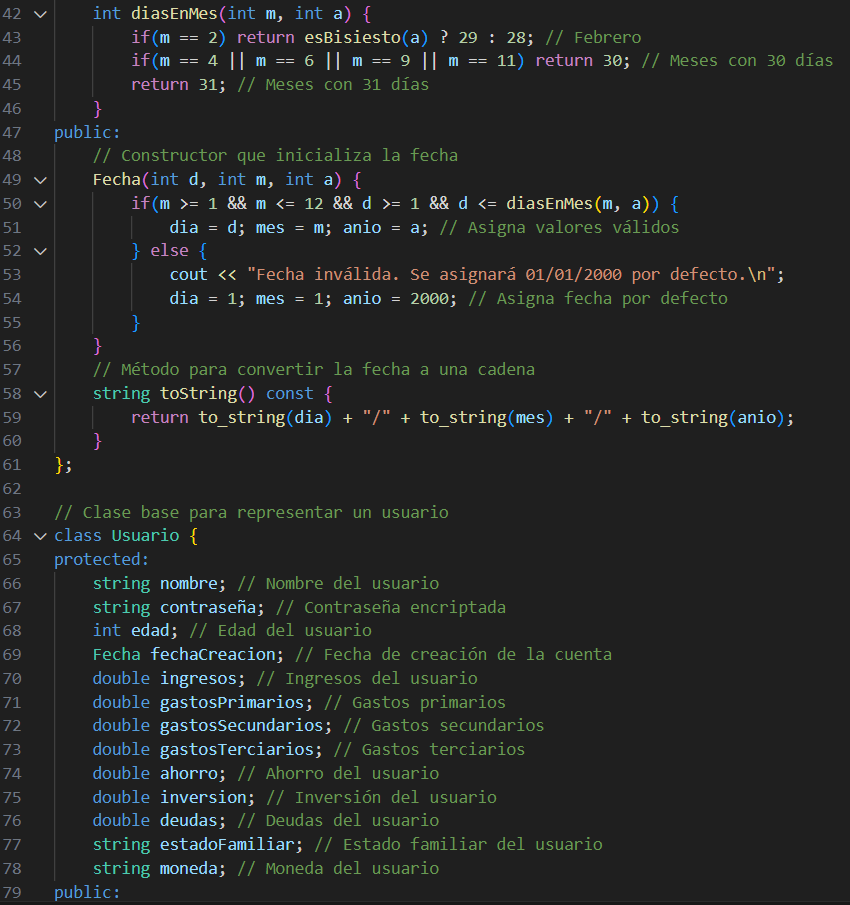
La inclusión de herramientas visuales, como la tabla de resumen y la barra de distribución financiera, refuerza la comprensión del manejo del dinero y ayuda a los usuarios a tomar decisiones informadas sobre ahorro, inversión y control de gastos. Estas funcionalidades promueven una educación financiera más efectiva, especialmente para aquellos que buscan mejorar sus hábitos de consumo y planificación económica.

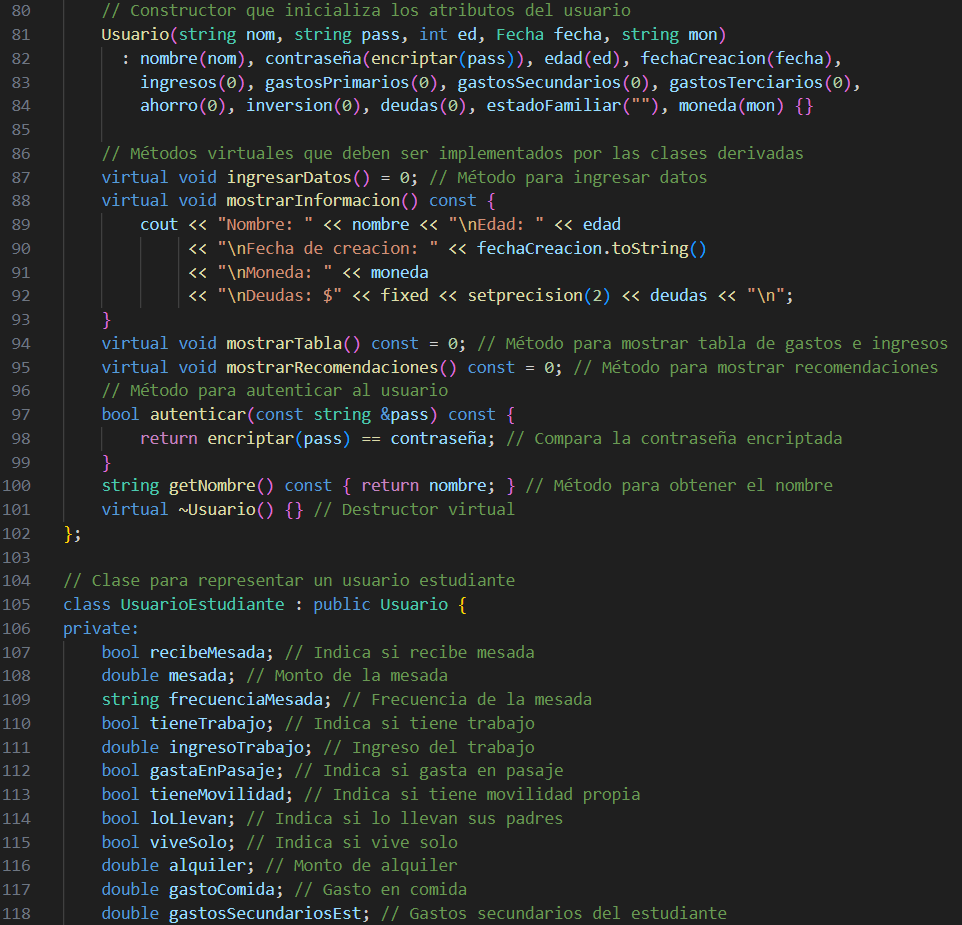
En conclusión, este código representa una solución integral y flexible para la gestión financiera personal, brindando a los usuarios las herramientas necesarias para administrar sus recursos económicos de manera organizada y eficiente. La combinación de programación estructurada, validaciones estrictas y reportes visuales permite que este sistema sea una base sólida para futuras mejoras y expansiones, adaptándose a nuevas necesidades y tecnologías.

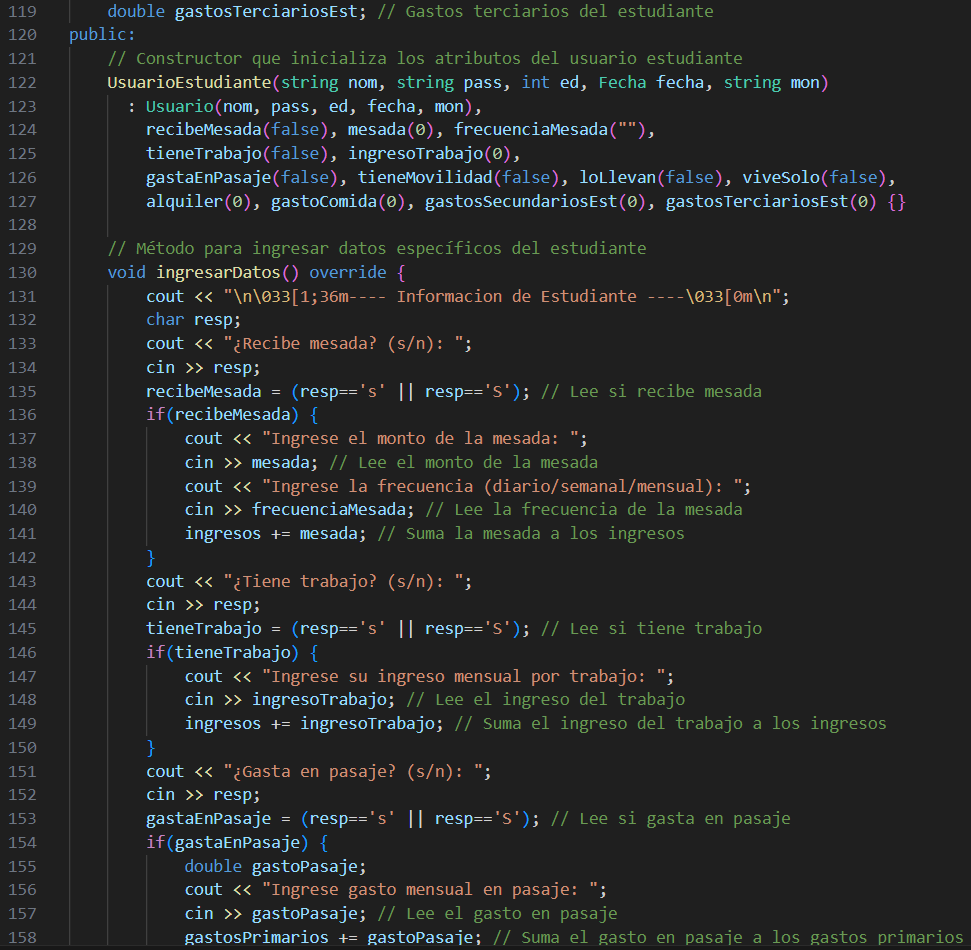
# **Anexos**

## **Código completo**

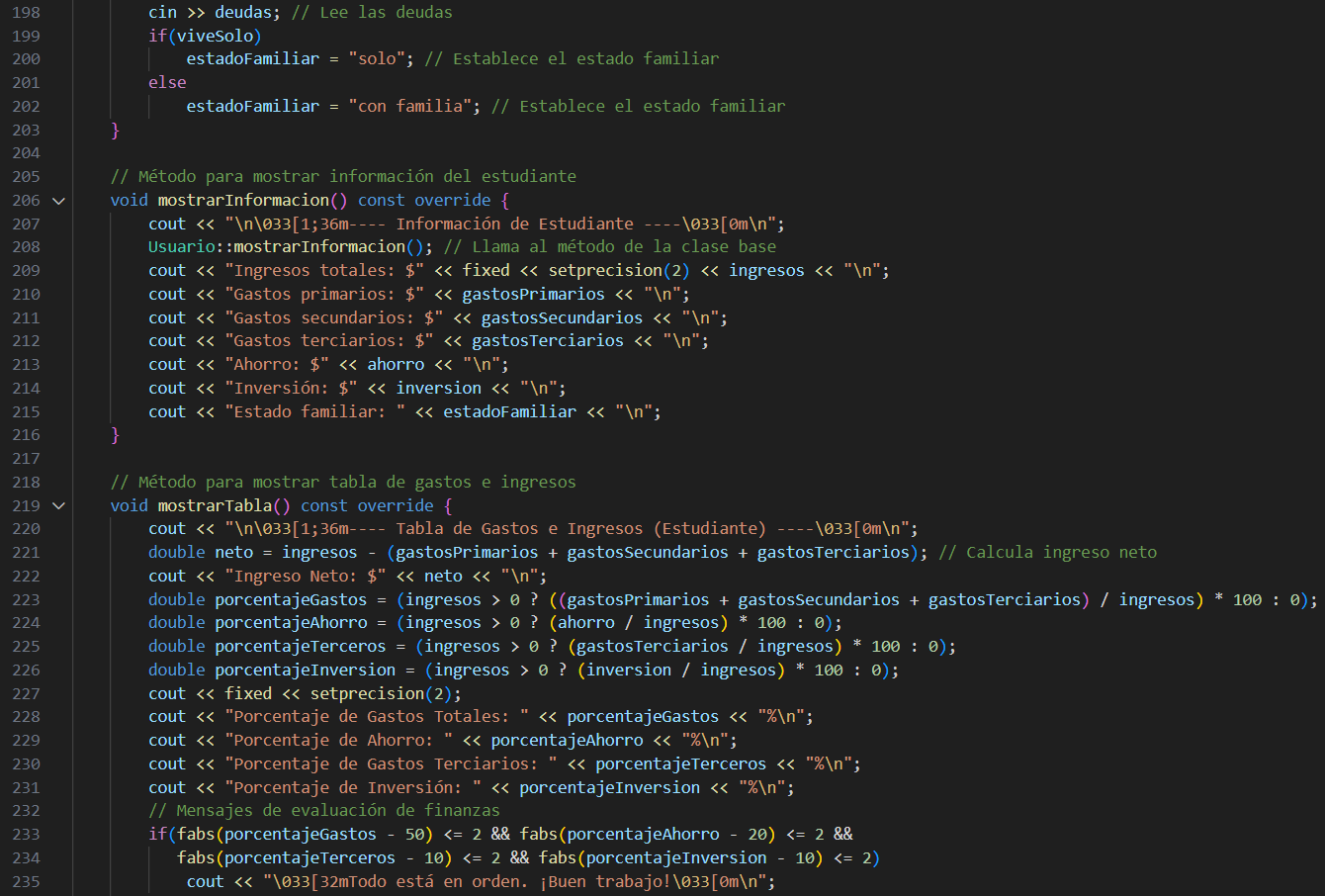
****

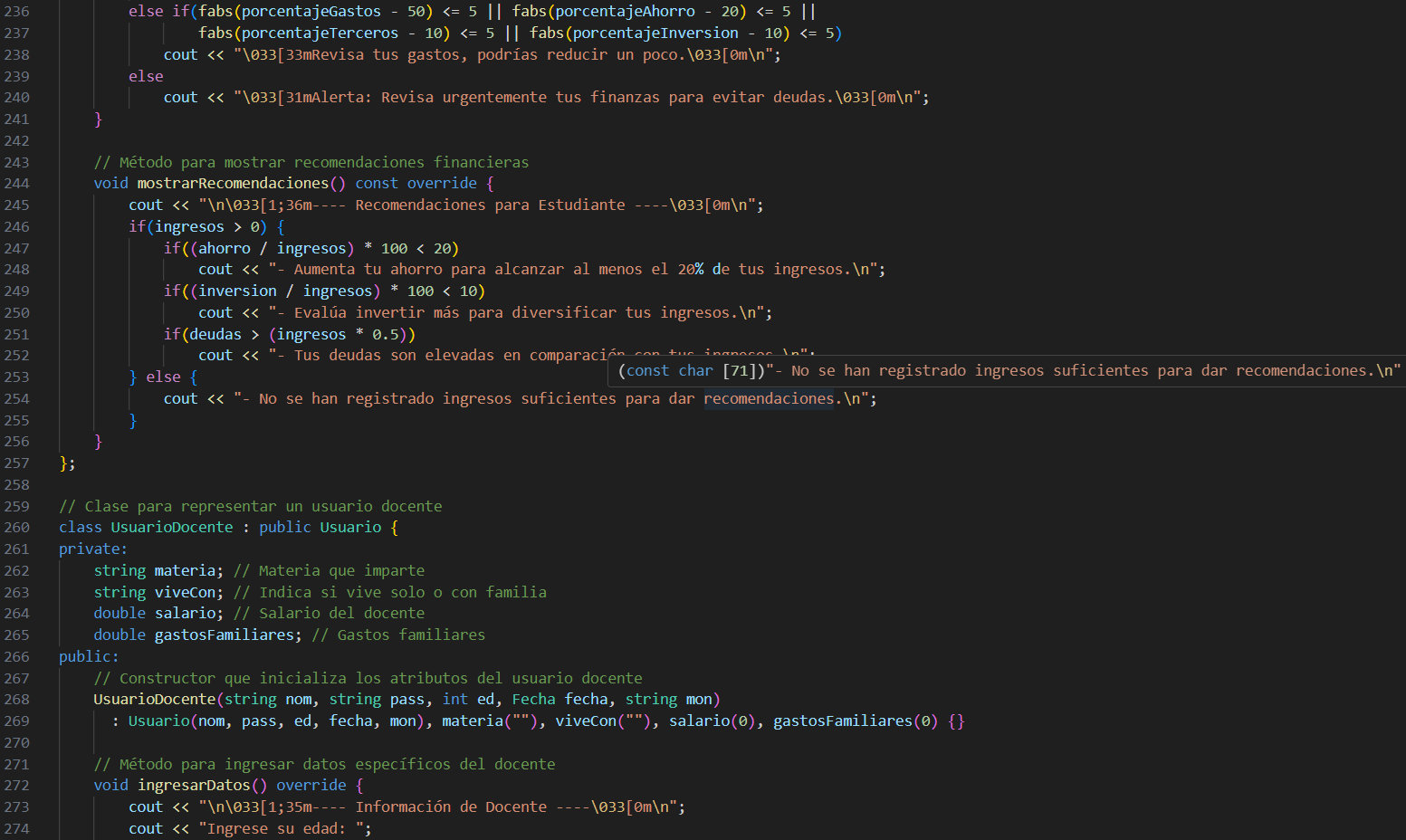
****

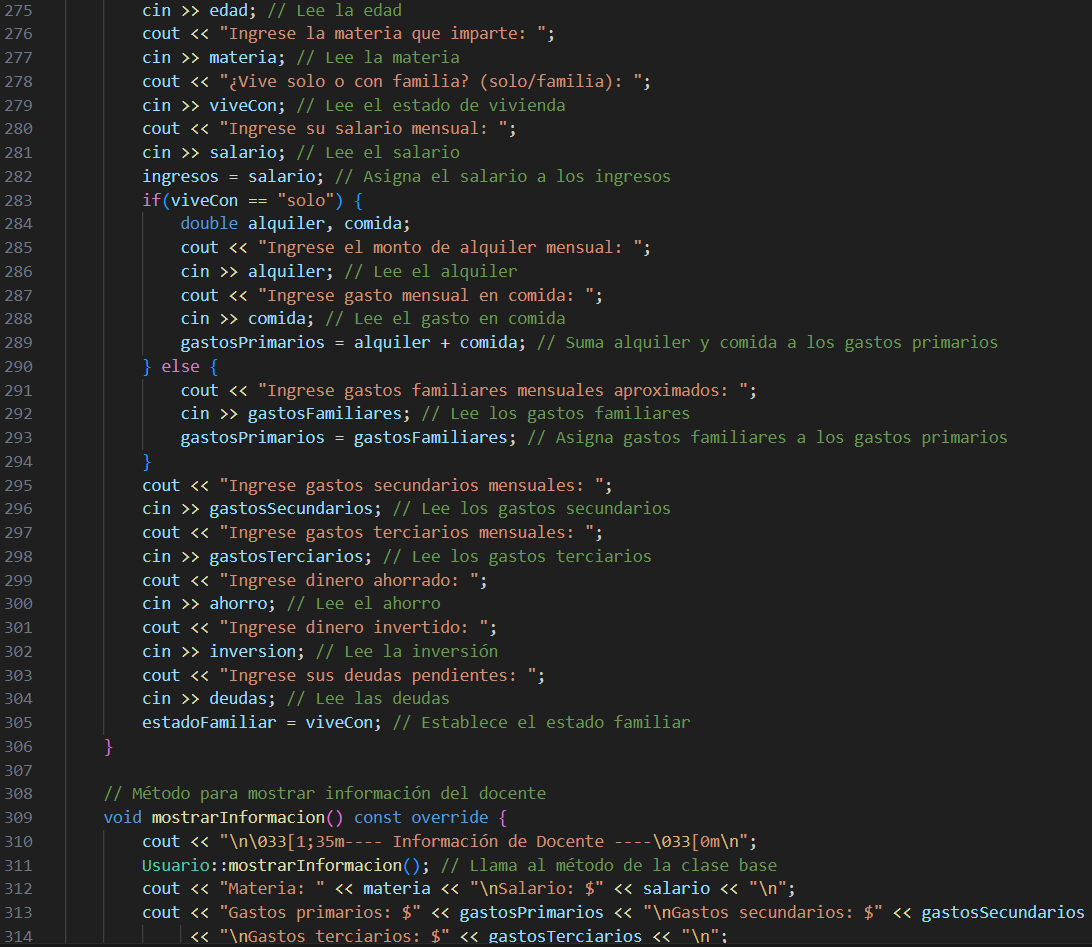
****

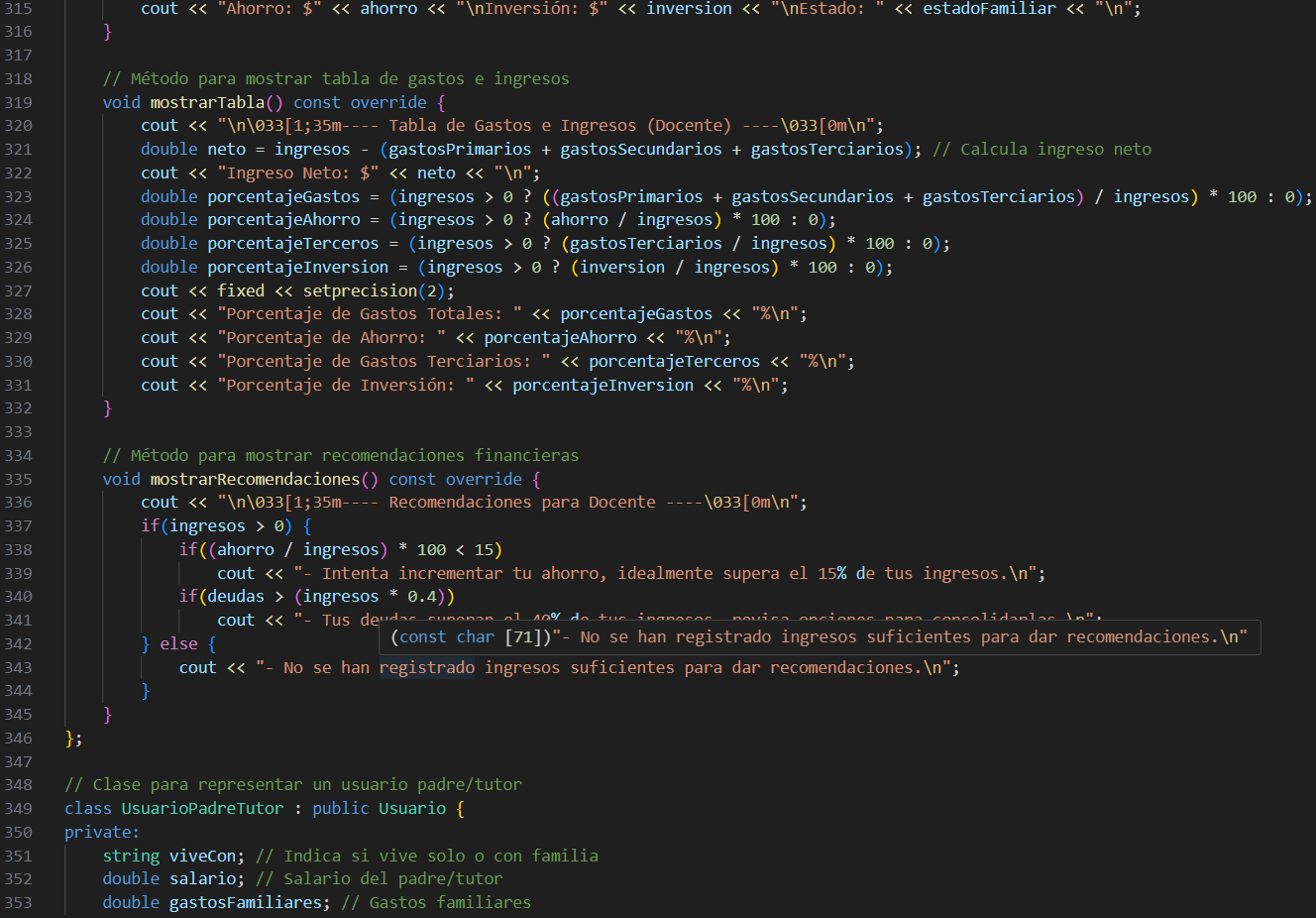
****

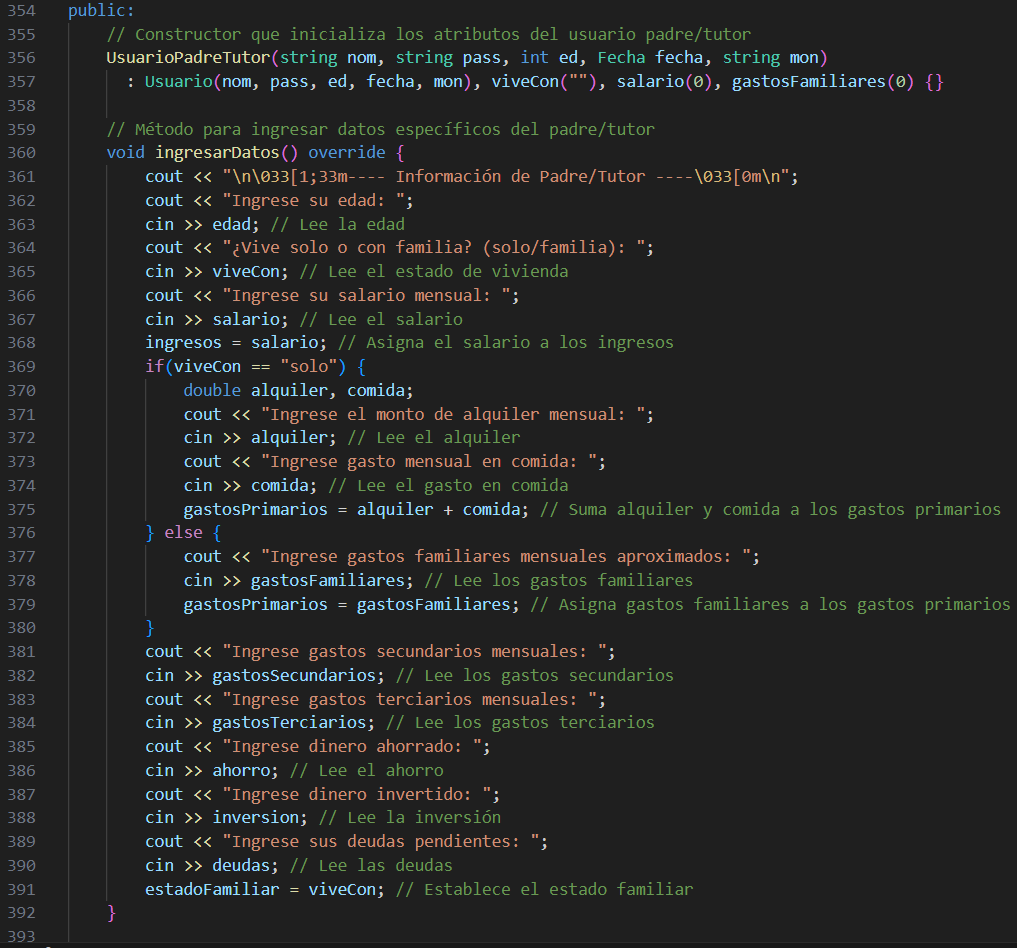
****

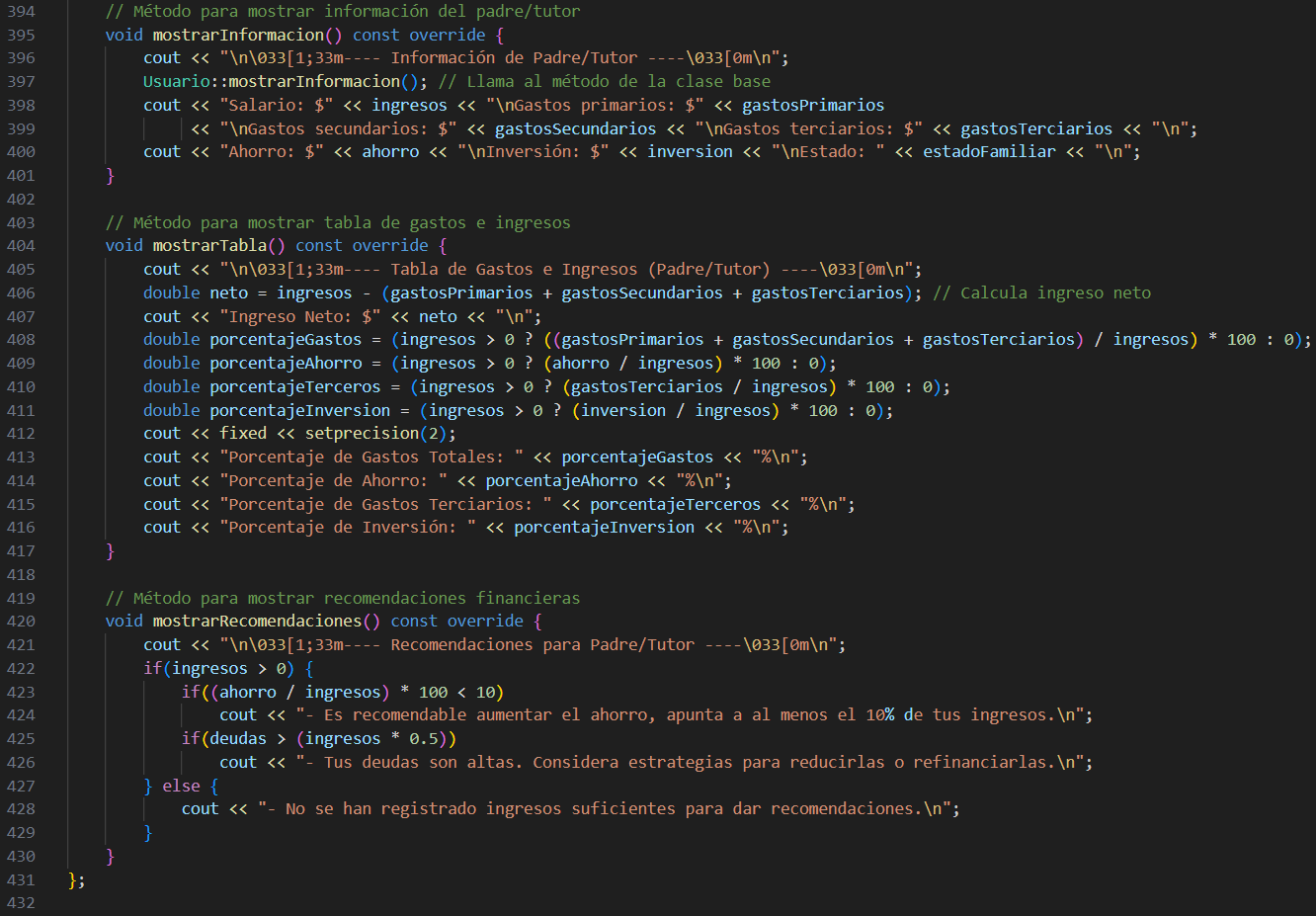
****

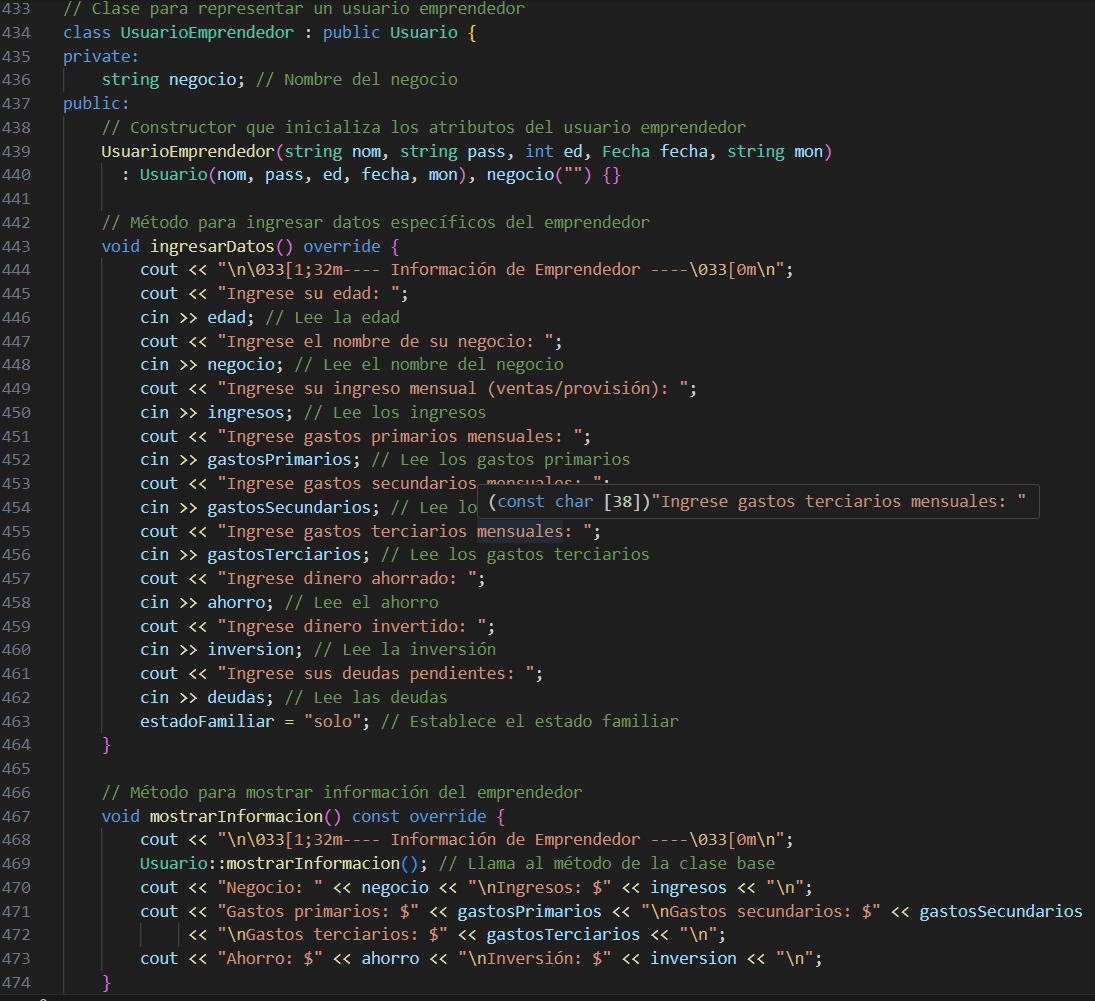
****

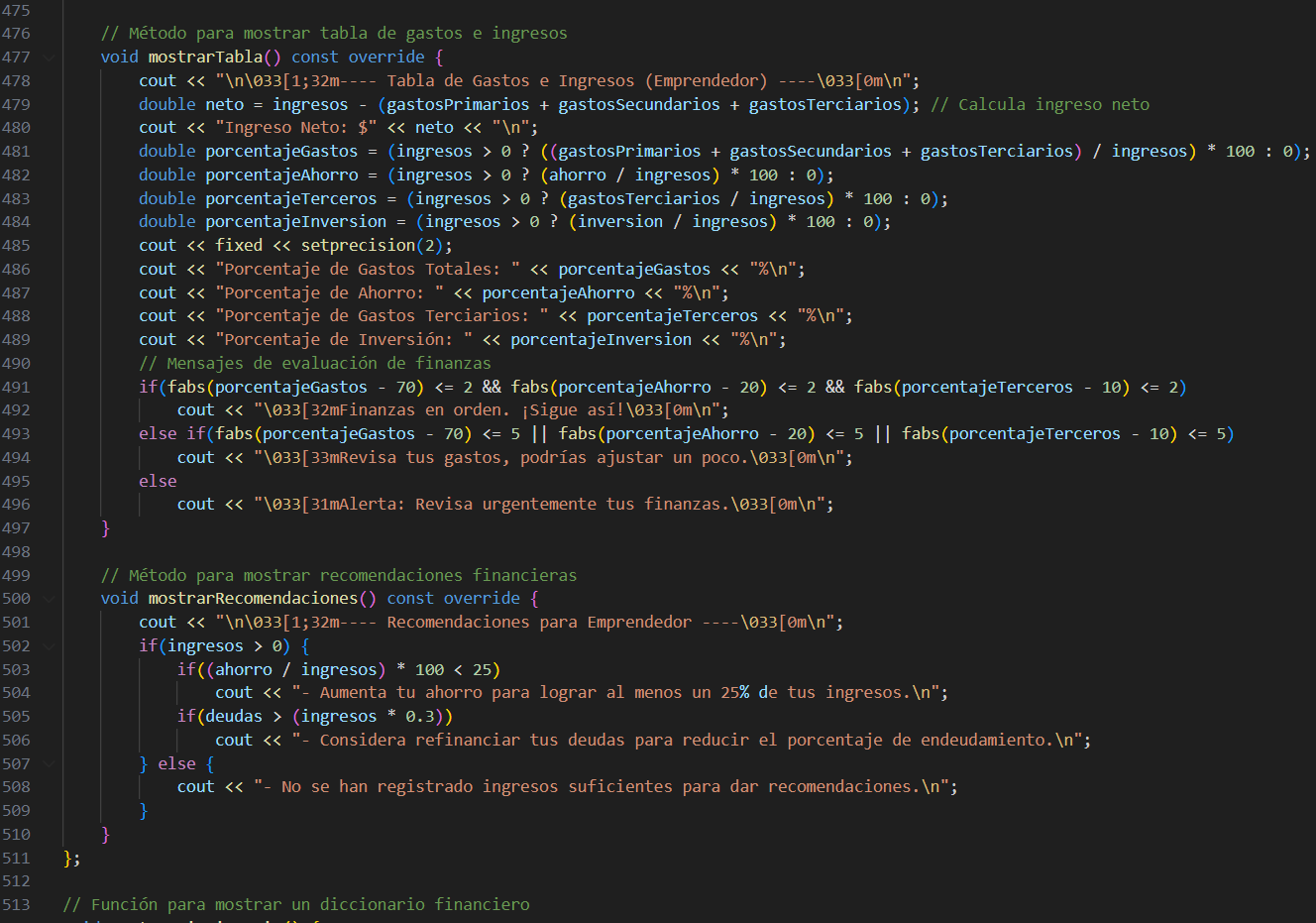
****

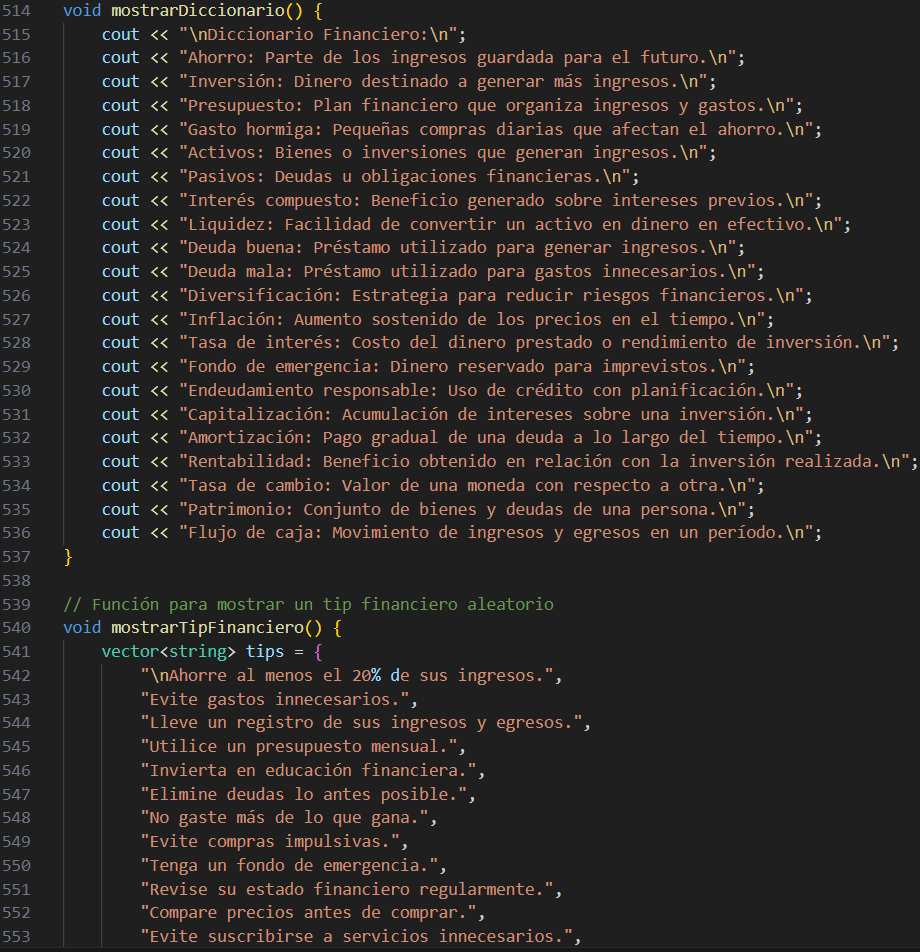
****

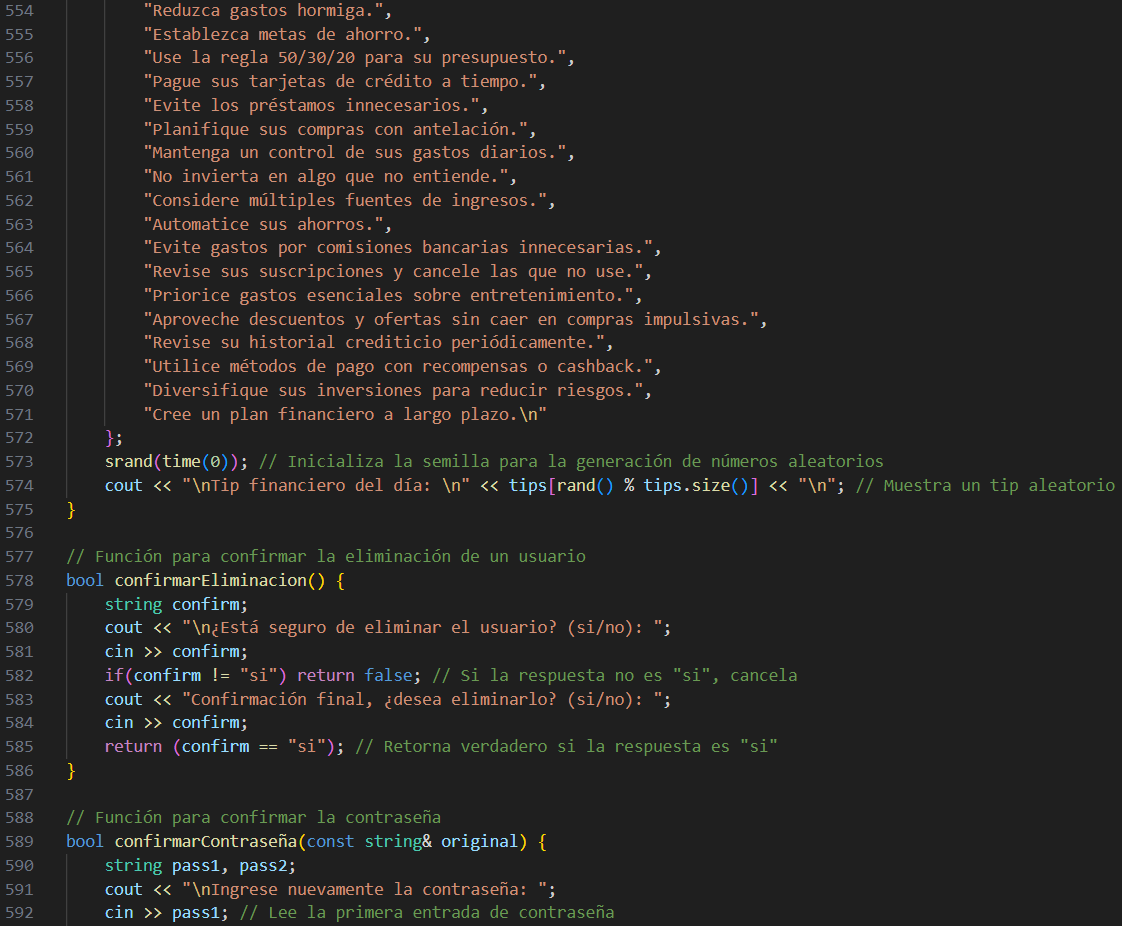
****

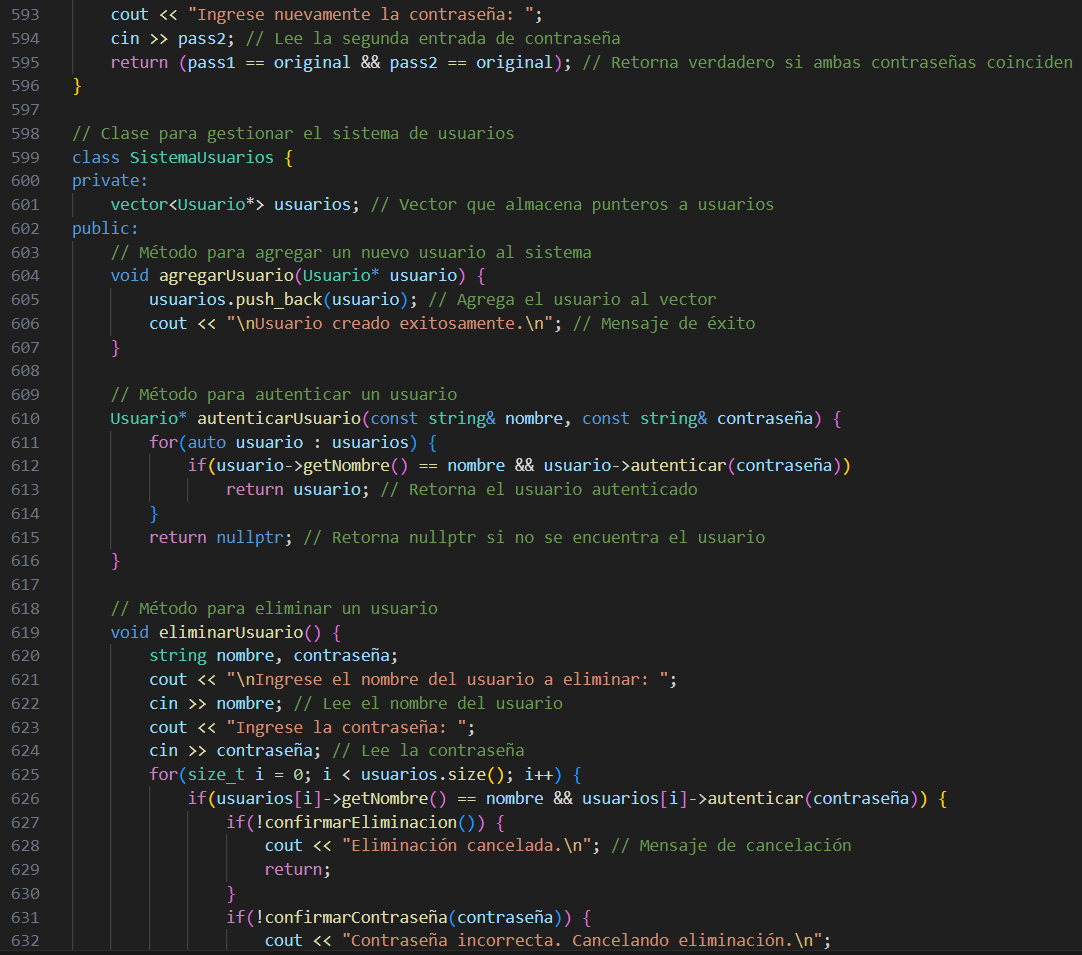
****

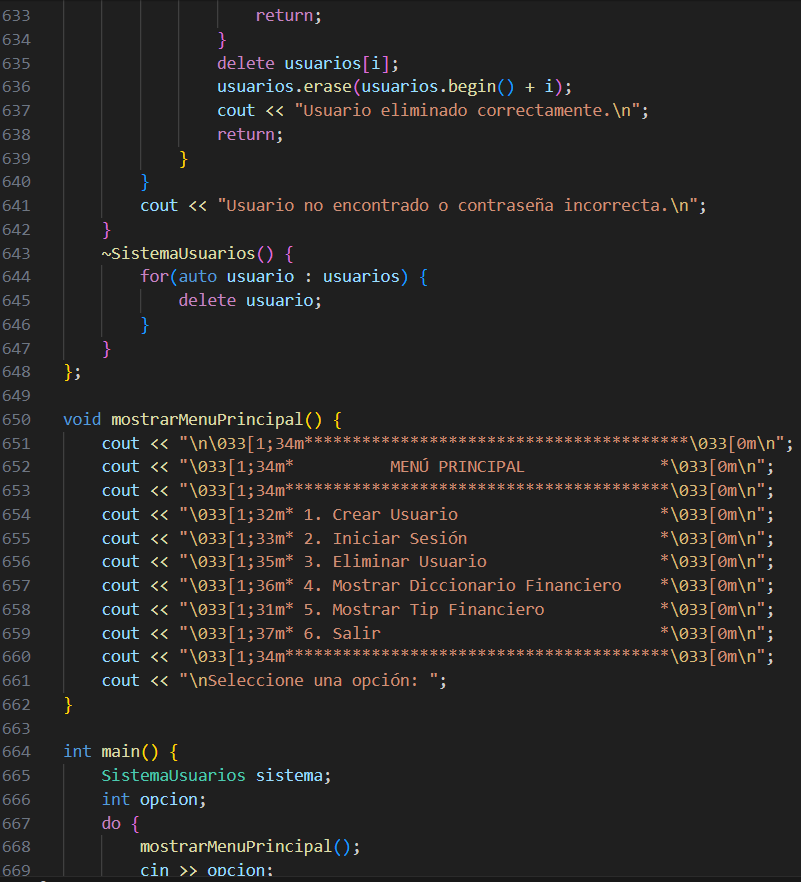
****

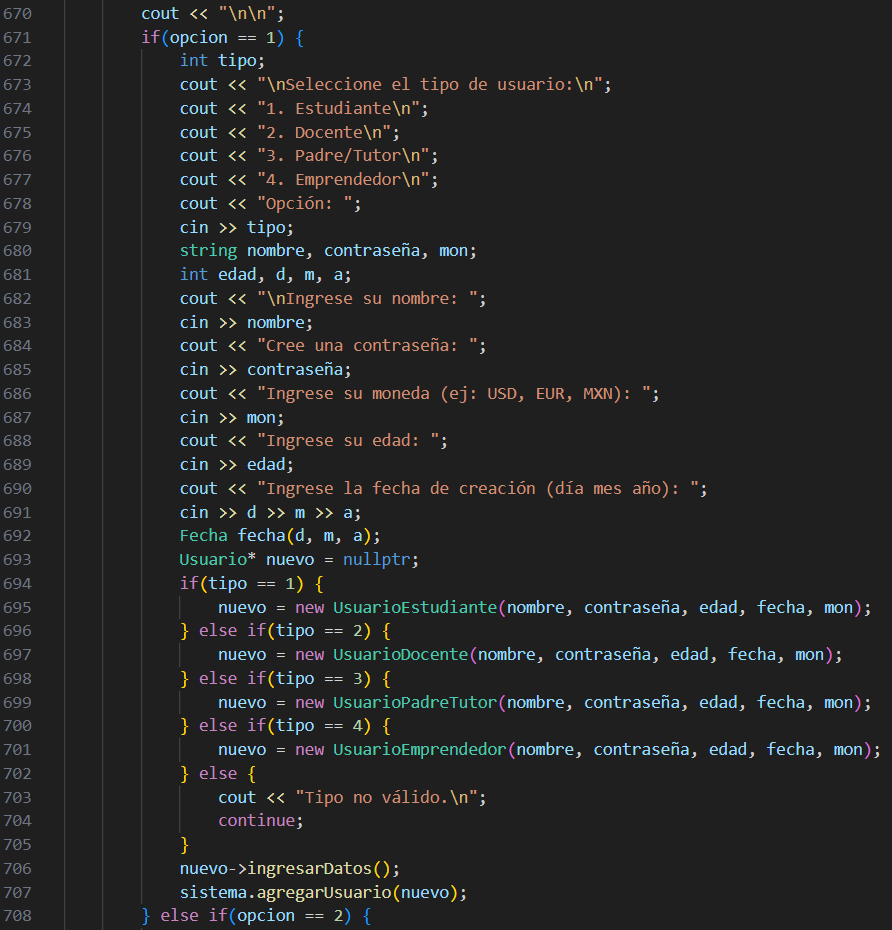
****

****

****

****

****

****

**Texto

Descripción generada automáticamente**

## **Diagrama**

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

## **URL Git**

## **Video en YouTube**