

INGENIERIA DE REQUERIMIENTOS Y DISEÑO DE SOFTWARE

**DISEÑO Y DESARROLLO DE SOFTWARE**

**CICLO III**

## Ciclo de Vida en Cascada

**Integrantes:**

**Docente:**

Gómez Marín, Jaime

**LIMA-PERU**

**MARZO DEL 2020**

**Índice**

* Caso práctico y enfoque metodológico..........................................3
* Requerimientos ...............................................................4
* Análisis ..........................................................................................5
* Diseño .............................................................................6-7
* Implementación .............................................................................8-9
* Pruebas ...........................................................................10-14
* Implantación .................................................................................15-16
* Bibliografía .....................................................................17

**Caso Práctico**

Se desea desarrollar un sistema de software para la SUNAT que me permita calcular el porcentaje de retención de los trabajadores de quinta categoría para el ejercicio del año pasado.



*Fuente:* [*http://orientacion.sunat.gob.pe/index.php/personas-menu/impuesto-a-la-renta-personas-ultimo/rentas-de-quinta-categoria-personas/3071-02-calculo-del-impuesto*](http://orientacion.sunat.gob.pe/index.php/personas-menu/impuesto-a-la-renta-personas-ultimo/rentas-de-quinta-categoria-personas/3071-02-calculo-del-impuesto)

**Desarrollo del Caso**

Para el desarrollo del proyecto de software, se empleará la metodología del ciclo de vida en cascada. Esta posee 6 fases, las cuales se irán desarrollando posteriormente.

- Requerimientos

- Análisis

- Diseño

- Implementación

- Pruebas

- Implantación

**Requerimientos:**

Una descripción detallada de cómo se han de cumplir los requisitos del proyecto.

* El software debe de estar en idioma español.
* Debe ejecutarse sin interfaz gráfica de usuario
* El lenguaje de programación debe ser Python
* Requerimientos funcionales:

- Debe permitir ingresar un primer monto (Remuneración mensual)

- Debe permitir ingresar un segundo monto (Gratificaciones)

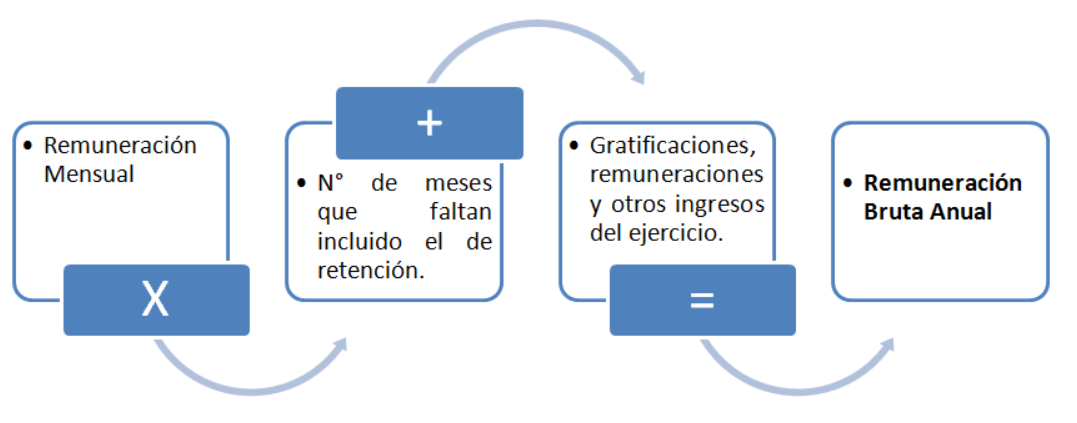
- Como resultado, se debe visualizar la remuneración anual bruta, remuneración anual neta y finalmente el impuesto anual proyectado.

Cálculo del Impuesto

1. Cálculo de la Remuneración Bruta anual

La **remuneración mensual** **se multiplica por 12 meses**. Luego, se le **suma las gratificaciones** para finalmente obtener la **Remuneración Bruta Anual**

Observación: Por limitaciones del software, el número de meses para el cálculo de retención serán 12.



1. Evaluación de la Remuneración Bruta Anual.

Si la "**Remuneración Bruta Anual" no supera las 7 UIT** entonces **NO ESTARÁ SUJETO A RETENCIÓN.** Caso contrario, se procederá con el cálculo de la Remuneración Neta Anual.

Nota: El valor de 1 UIT para el 2019 -> S/. 4200

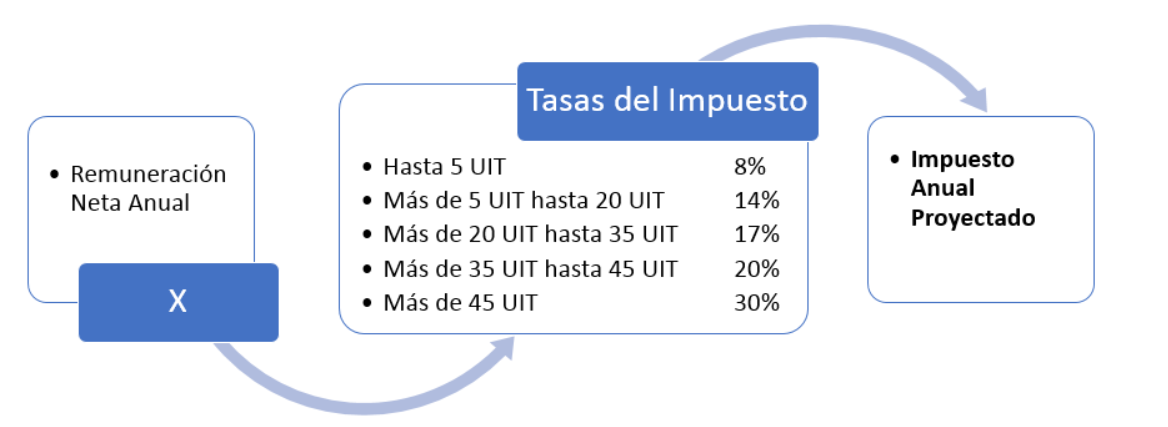
1. Cálculo de la Remuneración Neta Anual

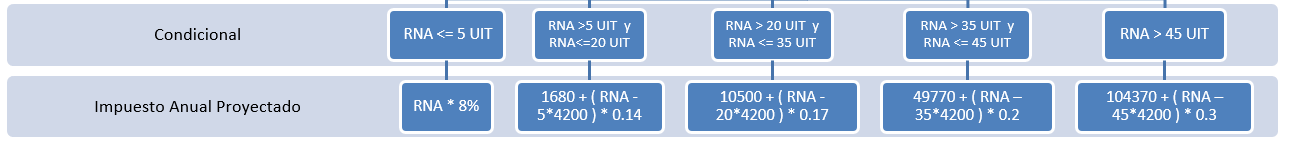
La Remuneración Bruta Anual se resta el valor de 7 UIT . Para obtener la Remuneración Neta Anual.



1. Calculo del Impuesto Anual Proyectado

De la **remuneración neta anual** , a efectos de obtener el impuesto anual proyectado, se deberá aplicar las siguientes tasas:





**Análisis:**

En esta etapa se consolida todo lo que se requiere del sistema y será aquello lo que seguirá en las siguientes etapas

**Diseño:**

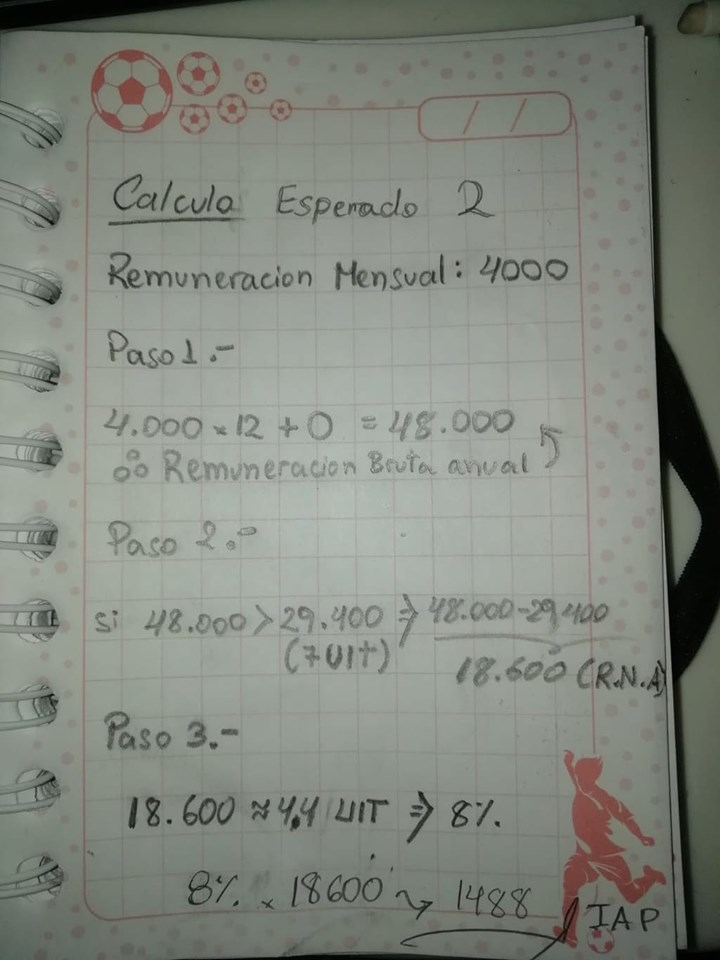
En esta fase se encarga de diseñar la **arquitectura de software**, así como un **plan de diseño** detallado del mismo.

**Arquitectura del Software**

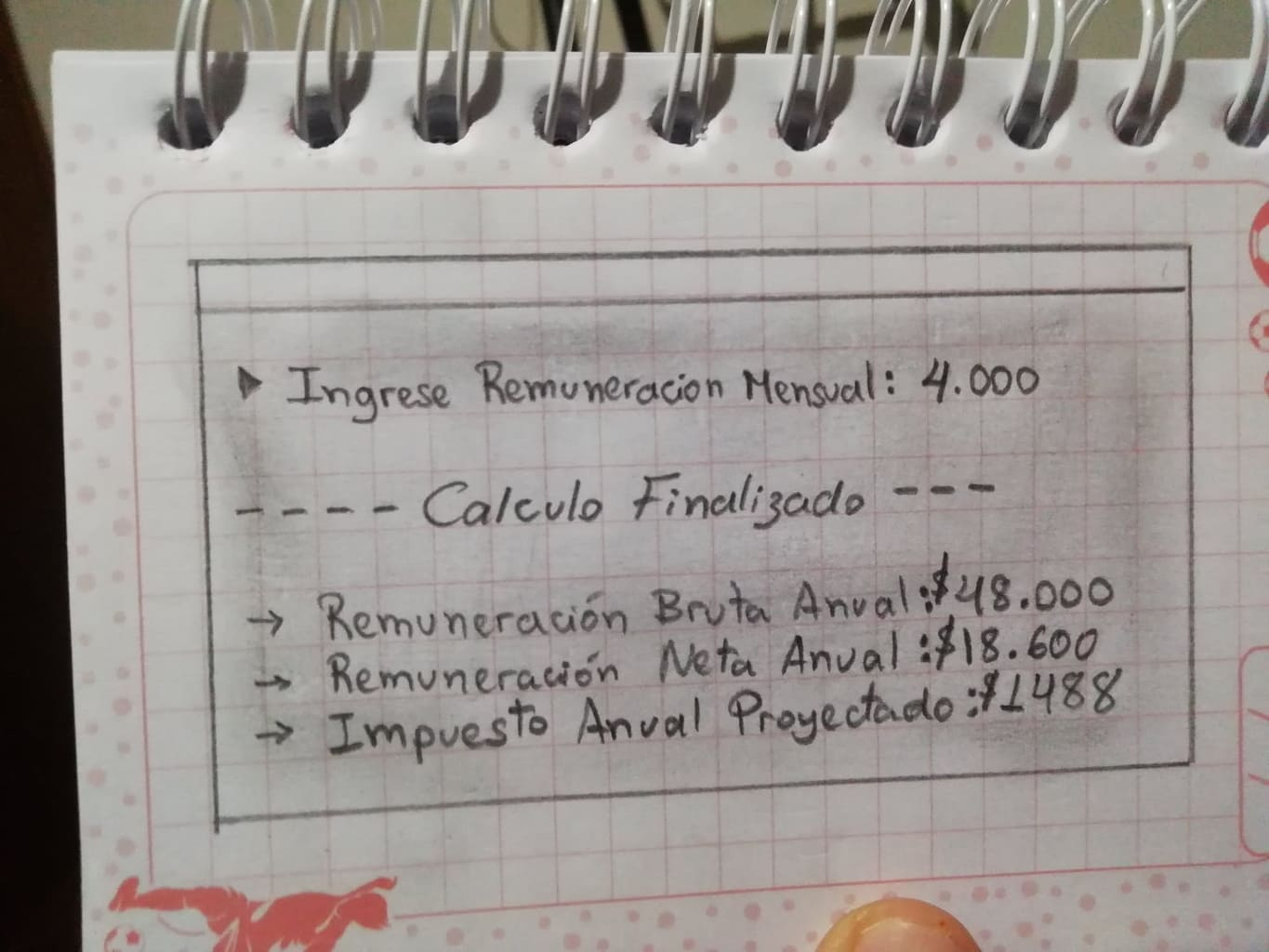
Lógica

**Plan de Diseño (Bocetos)**

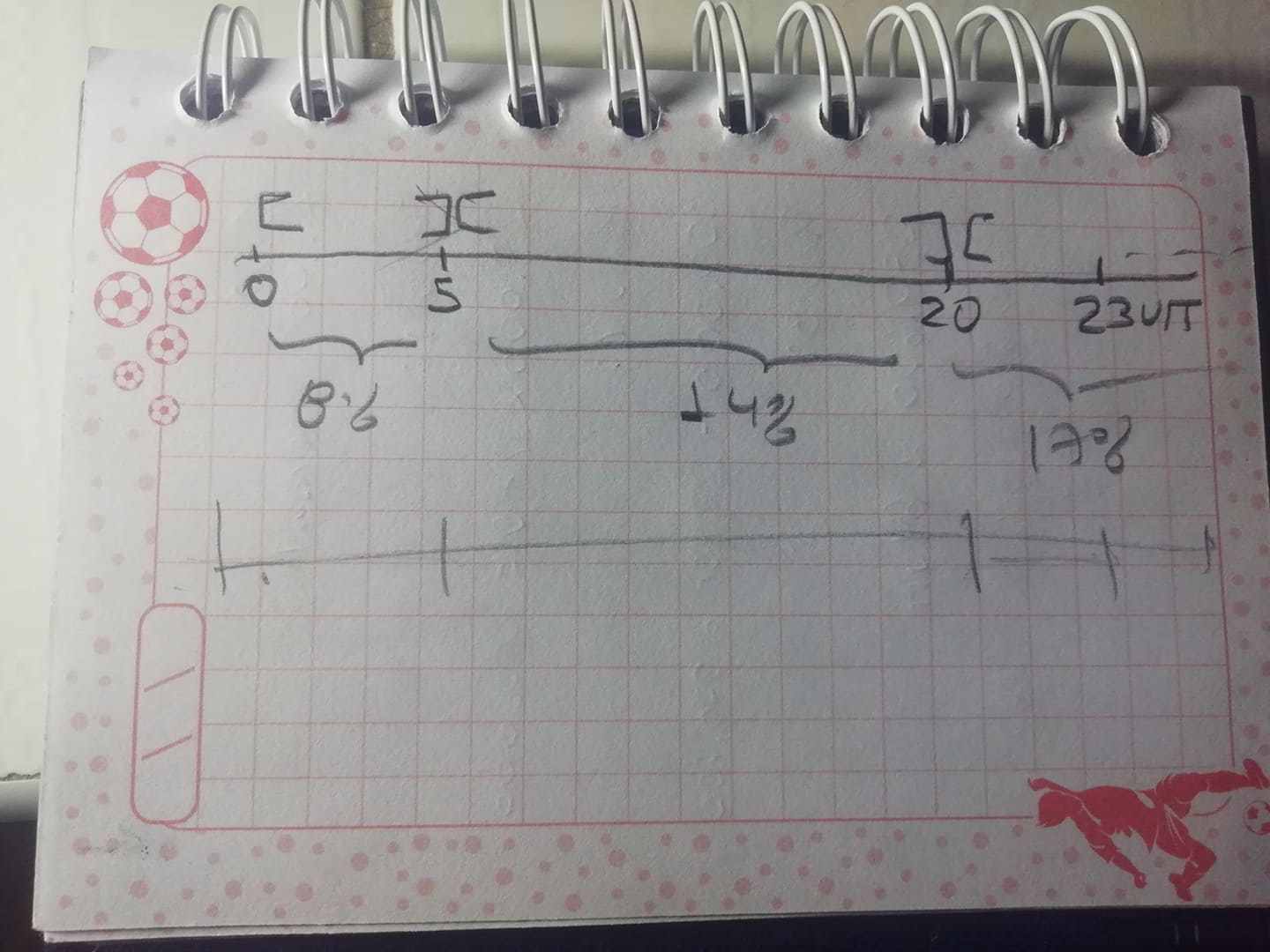
Cálculos Internos (Código)



Emulador de Terminal (Interfaz)



Cálculos Internos (Código)

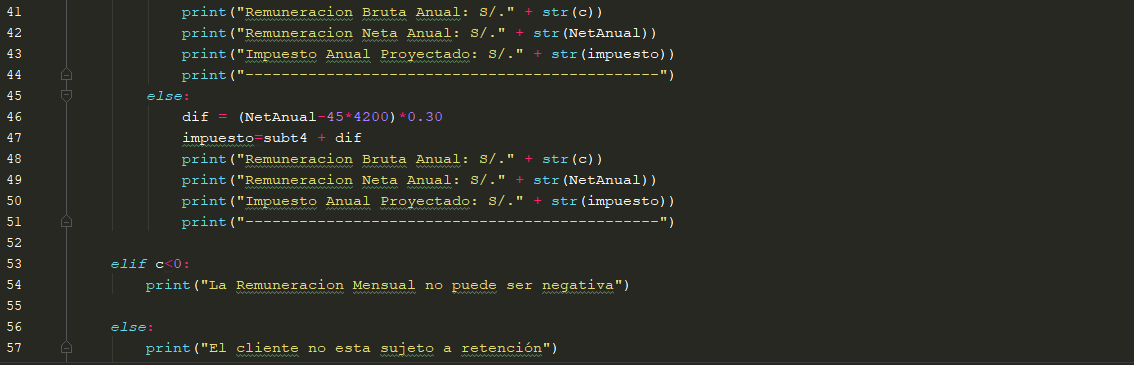


**Implementación:**

Lógica de los requisitos.

Código fuente. (Lenguaje de programación : Python )

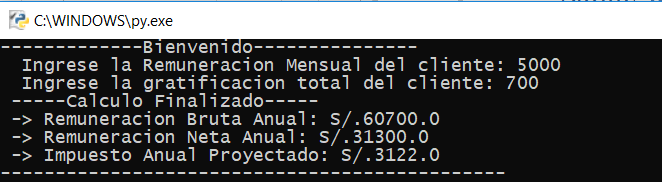


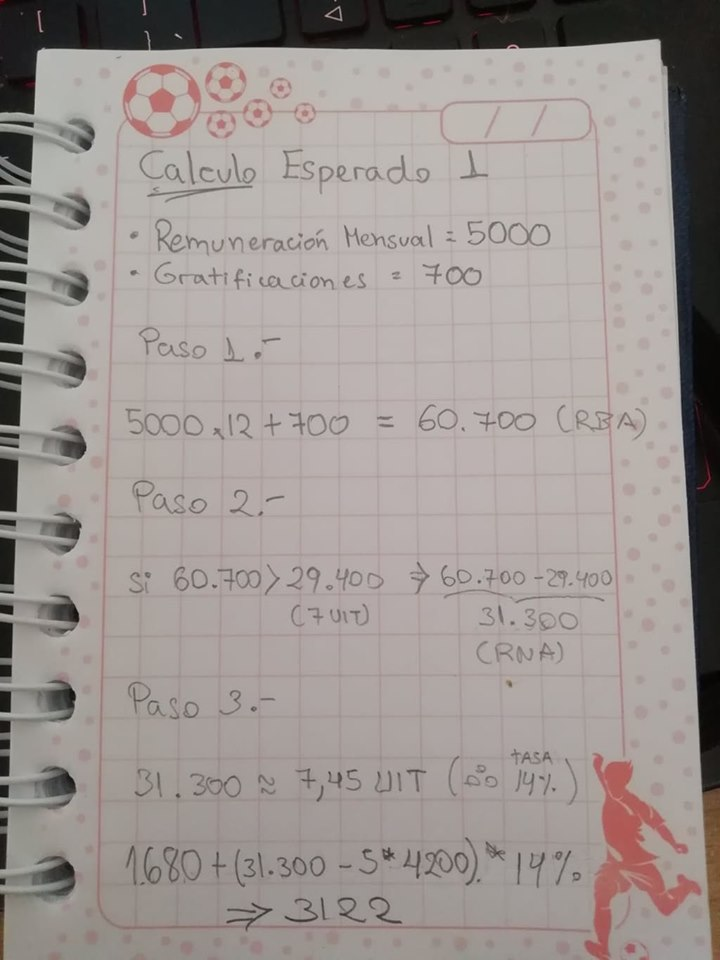


**Prueba:**

Las pruebas permiten determinar si el software cumple con las exigencias definidas con anterioridad.

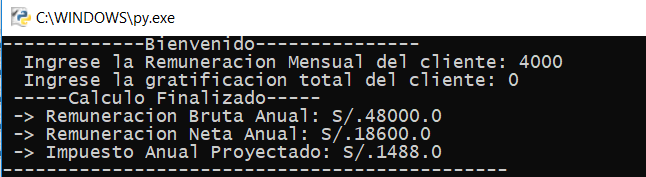
Prueba 1:

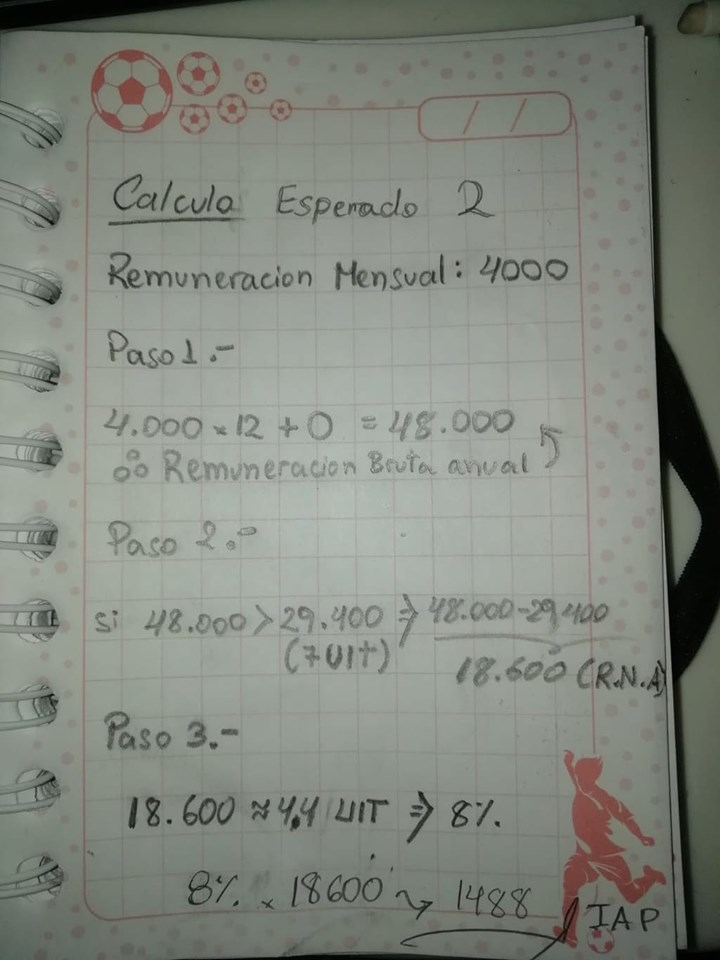




**CÁLCULO ESPERADO**

Prueba 2:

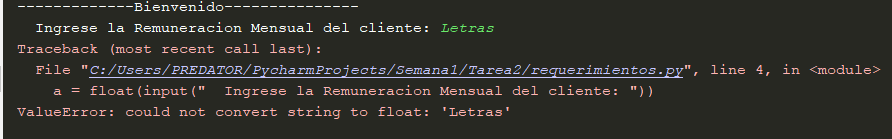




**CÁLCULO ESPERADO**

Prueba 3:

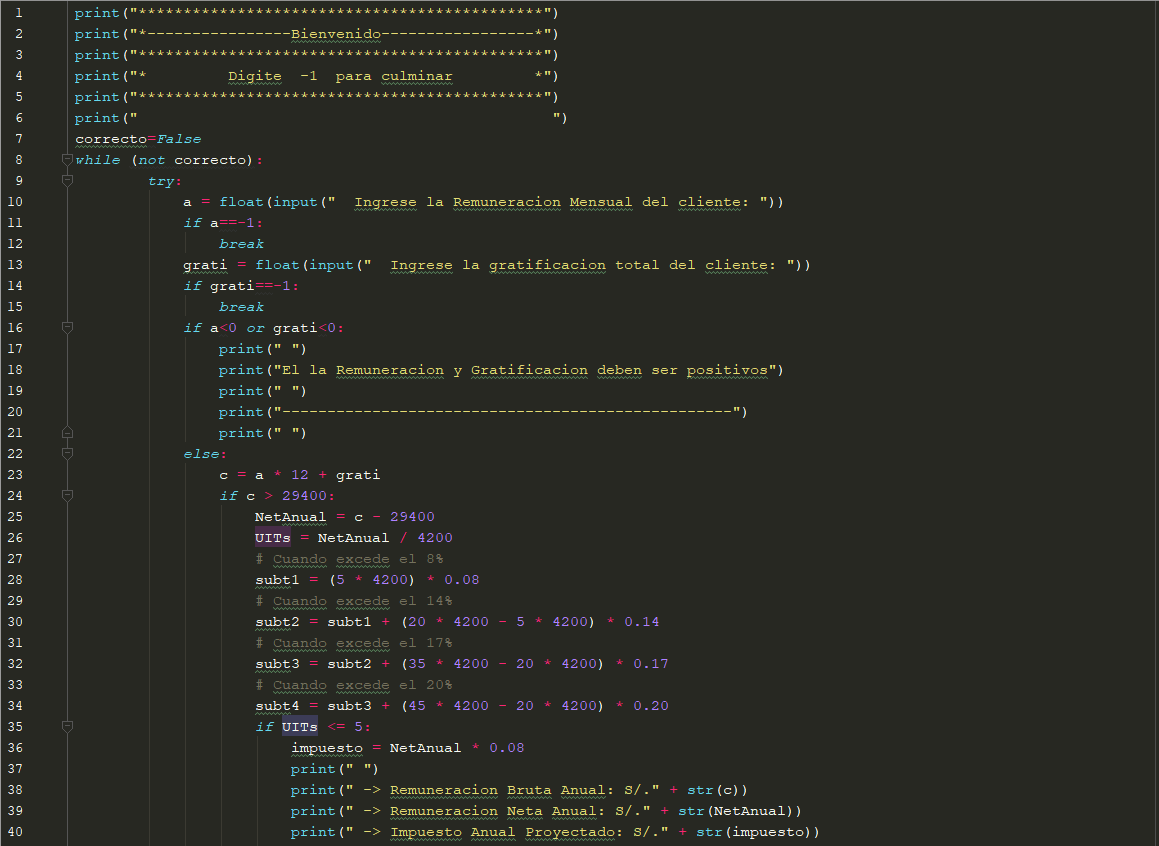
(Nota: Se ejecutó el software en la consola de Python para apreciar el error en detalle)

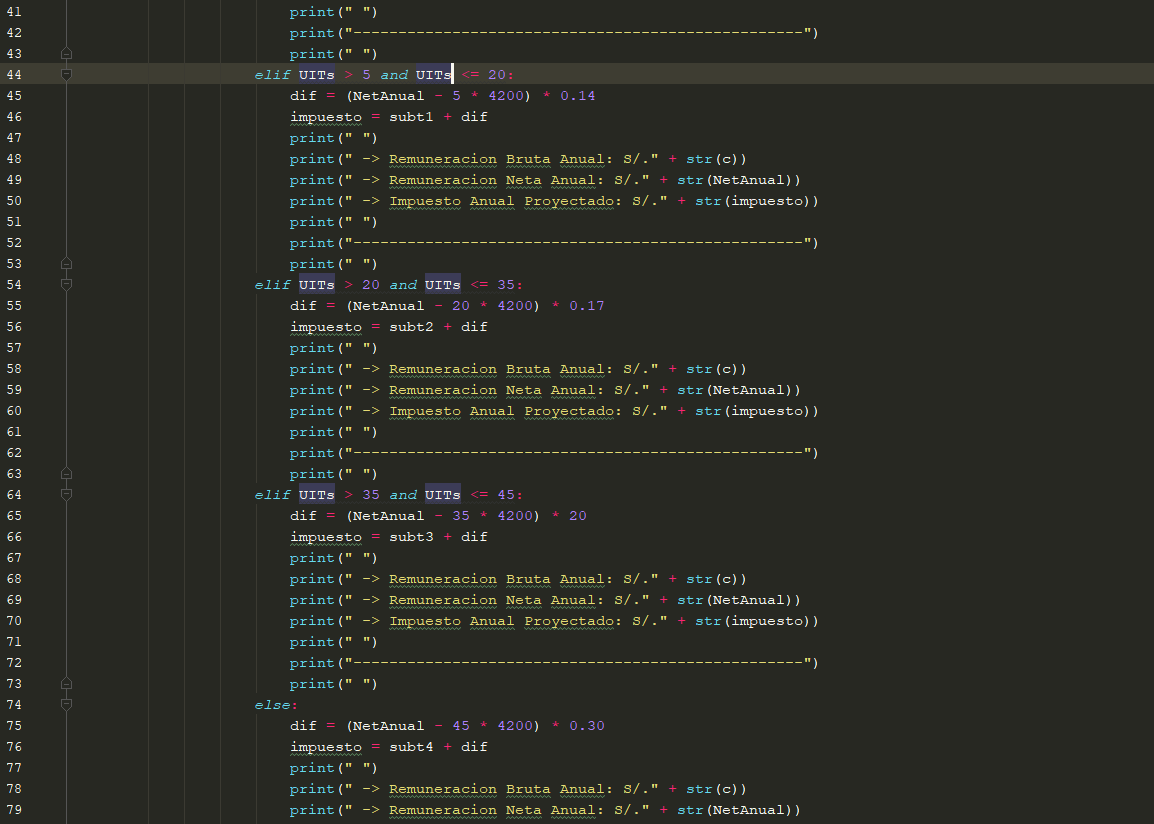


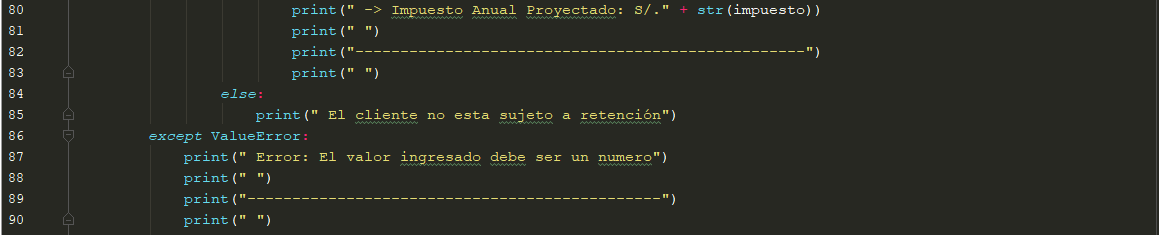
**ERROR EN EJECUCIÓN**

Los cálculos realizados por el software son los esperados. Sin embargo, faltó implementar excepciones a los posibles errores en ciertas instrucciones del programa. Así como también, un método que permita terminar el proceso de ejecución del mismo.

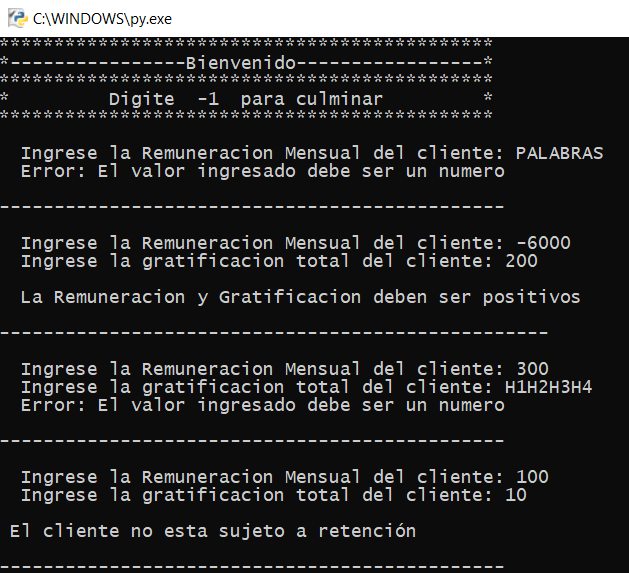
Código fuente actualizado.



**

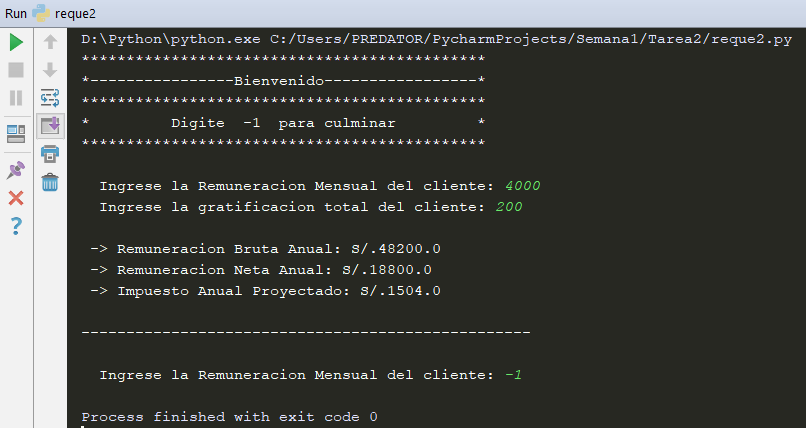


Prueba 4: (Manejo de excepciones)



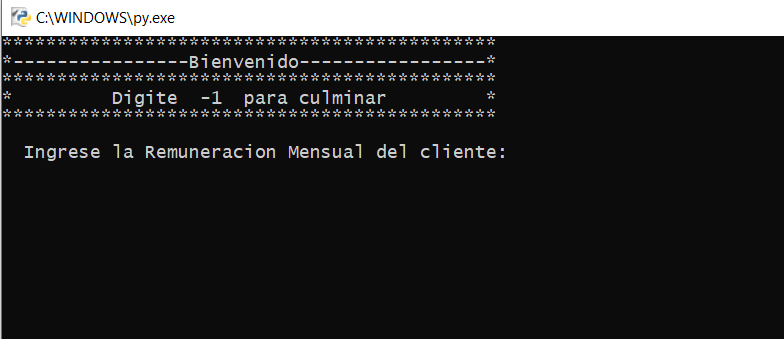
Prueba 5 ( método de salida )

( Se optó por el numero “ -1 “ para culminar el proceso )



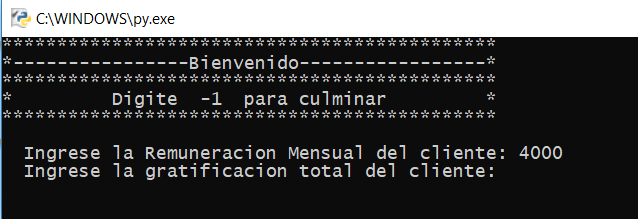
**Implantación:**

1. Encabezado: Primero muestra la bienvenida al usuario y un recordatorio. Luego, da a conocer que el número -1 se emplea para culminar el proceso.



**Encabezado**

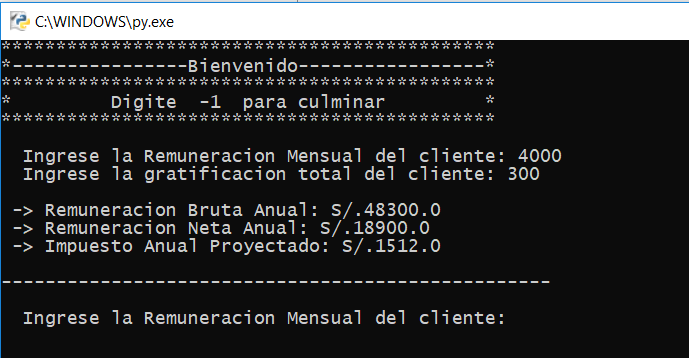
1. Requisitos: Son la remuneración mensual y la gratificación. Se ingresan respectivamente en la terminal, y son necesarias para obtener los resultados deseados.



**Requisitos**

1. Respuestas: Aparecen Inmediatamente después de ingresar la gratificación (ultimo requisito).

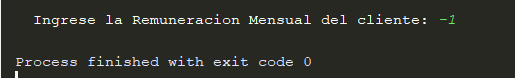
Después de mostrar los resultados, aparece una línea divisora, el cual indica que el proceso anterior ha culminado y ya se pueden ingresar nuevos requisitos.



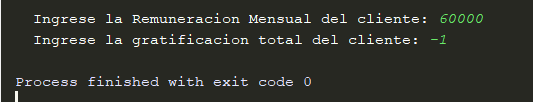
**Respuestas**

1. Para terminar el proceso de ejecución se digita el numero -1 ,ya sea en el primer o segundo requisito.

(ejecutado desde el terminar de Python)



(ejecutado desde el terminar de Python)



Nota: Ejecutado desde consola, el terminal se cierra.

**Bibliografía:**

# University of Missouri. (2016). The Traditional Waterfall Approach. Comparing

# Traditional Systems Analysis and Design with Agile Methodologies. Columbia, EEUU. Recuperado de http://www.umsl.edu/~hugheyd/is6840/waterfall.html

Schwaber, Steve. (2011). SDLC - Waterfall Model. Tutorialspoint

simplyeasylearning. Recuperado de

http://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm

Domínguez, Pablo (2016). En qué consiste el modelo en cascada*. Gestiona*

*tu proyecto de desarrollo*. Recuperado de https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada