El proyecto final que decidí realizar fue el de la aplicación de finanzas personales puesto que me vendría bien un aplicativo de este tipo para los tiempos que corren.

Supuestos:

* Interfaz/interactividad con el usuario: El usuario identifica la funcionalidad de la aplicación fácilmente puesto que por medio de etiquetas con frases clave tales como: “Enter Expenses”, “Save in …”, “Notes:”, “Inputs”, “Registry”, así como otras que especifican el tipo de información contenida en sus respectivos campos, se usaron colores usualmente relacionados con algo que es beneficioso (verde) y algo que no los es tanto (rojo), se hace uso de botones para archivar los datos ingresados los cuales son codificados y guardados en archivos de código binario, esta información previamente guardada se muestra automáticamente en el apartado llamado registro, esto para llevar un seguimiento más detallado y al instante de la información que se está tratando, en este ultimo apartado se hace uso de un botón cuya funcionalidad es hacer el calculo del balance mensual, también el guardado final de la información y el reseteo de la misma para dejarla completamente en blanco esto a través de un par de botones. También hay un pequeño apartado de texto en el cual el usuario puede ingresar notas de texto las cuales son permanentes siempre que el usuario lo decida así, también se adicionó un botón que despliega una calculadora sencilla en una ventana externa pero vinculada con la ventana principal. También se implementaron alertas que informan al usuario de los cambios que va llevando en el registro.
* Back-end: se usó la librería tkinter para la creación de la interfaz así como la librería pickle para la permanencia de los datos, se manejaron algunas excepciones en los que respecta al ingreso correcto de la información a través de la función “def is\_number(parameter\_1):” la cual retorna valores booleanos de acuerdo al tipo de información que el usuario intente adicionar siendo cualquier String de tipo no numérico el que lance la excepción, respecto a las demás operaciones posibles en la aplicación también se llevan a cabo a través de funciones , entre estas operaciones están: sumas de gastos e ingresos así como el balance realizado mensualmente por las funciones “def display\_earns\_data():, def display\_expenses\_data():” el guardado de información a través de la función: “def save\_db():” el reseteo de la información contenida en los archivos de datos de la aplicación se efectúa por medio de la función “def reset\_db():” la cual alerta al usuario del procedimiento y le pregunta si quiere continuar, en caso de que la respuesta sea afirmativa se lanzan dos funciones más “def undone\_expenses\_reset():, def undone\_earns\_reset():” las cuales se encargar de dejar limpia la pantalla de la GUI y borrar la información contenida en los archivos de datos, en lo que respecta a la función “def calculator\_window():” simplemente lanza en una ventana suelta una calculadora sencilla y con algunos “errorcitos” pero pues, no graves. Esta calculadora decidí adicionarla puesto que al ser una aplicación de finanzas convendría bastante tener a la mano y de manera muy inmediata una calculadora.