Universidad Tecnológica de Durango

Tecnologías de la Información

Programación Orientada a Objetos (POO)

Proyecto Final

*“Gestor escolar”*

Alumnos:

* Rodriguez Alarcón Ricardo
* Tinoco Sánchez Alexander

2°B

Docente:

* Ing. Dagoberto Fiscal Gurrola, M.T.I.

Durango, ago. 2024

Tabla de contenido

[Tabla de ilustraciones 1 3](#_Toc174887929)

[Objetivos 5](#_Toc174887930)

[General 5](#_Toc174887931)

[Específico 5](#_Toc174887932)

[Descripción del problema 6](#_Toc174887933)

[Diagrama relacional de base de datos 7](#_Toc174887934)

[Diagrama UML de clases y objetos 8](#_Toc174887935)

Capturas de la codificación……………………………………………………10-21

Capturas de la ejecución………………………………………………………21-31

[Conclusiones 32](#_Toc174887941)

[Bibliografía 33](#_Toc174887942)

# Tabla de ilustraciones 1

Ilustración diagrama relacional de base de datos……………………………………7

Ilustración Diagrama de objetos y clases……………………………………………8

Ilustración 1 conexión a MYSQL…………………………………………………..10

Ilustración 2 funciones 1……………………………………………………………11

Ilustración 3 clases 1………………………………………………………………..12

Ilustración 4 clases 2…………………………………………………………..……13

Ilustración 5 login y menús estudiantes……………….……………………..……14

Ilustración 6 menú estudiantes………………………………………………..……15

Ilustración 7 ventana profesores……….……………………………………..……16

Ilustración 8 insertar calificaciones………….………………………………..……17

Ilustración 9 agregar maestros y personas……………………………………..……18

Ilustración 10 main modificar alumnos………..……………………………..……19

Ilustración 11 modificar calificaciones………………………………………..……20

Ilustración 12 actualizar calificaciones………………………………………..……21

Ilustración 13 ejecución 1……………………………………………………..……22

Ilustración 14 ejecución 2……………………………………………………..……23

Ilustración 15 ejecución 3 ………..…….……………………………………..……24

Ilustración 16 ejecución 4……………………………………………………..……25

Ilustración 17 ejecución 5………………………………………………………….26

# Tabla de ilustraciones 2

Ilustración 18 ejecución 6……………………………………………………..……27

Ilustración 19 ejecución 7……………………………………………………..……28

Ilustración 20 ejecución 8 ………..…….……………………………………..……29

Ilustración 21 ejecución 9……………………………………………………..……30

Ilustración 22 ejecución 10………………………………………………………...31

# Objetivos

## General

* El propósito general de la materia de Programación Orientada a Objetos (POO) es proporcionar a los estudiantes las habilidades y conocimientos necesarios para desarrollar aplicaciones de software eficientes utilizando el paradigma de orientación a objetos. Esto implica aprender a diseñar, implementar y mantener soluciones de software que sean modulares, reutilizables y fáciles de entender.

## Específico

* Desarrollar un sistema de gestión escolar que permita agregar y modificar datos de alumnos y maestros. Además, los maestros podrán añadir y cambiar las calificaciones de sus alumnos. En este caso, el sistema fue creado para apoyar al Centro de Bachillerato Tecnológico Industrial y de Servicios Número 110.

# Descripción del problema

En el Centro de Bachillerato Tecnológico Industrial y de Servicios Número 110 se necesita de un sistema de gestión escolar donde se puedan agregar alumnos asi como modificar sus datos por si en algún caso los datos llegaran a necesitar alguna modificación. También se requiere una opción para agregar profesores al sistema, y al igual que los alumnos, también se agregó una opción para poder modificar sus datos si es necesario. Tanto los maestros como los alumnos serán capaces de modificar sus contraseñas a diferencia de los alumnos los profesores podrán modificar y agregar calificaciones.

Para que solo los maestros puedan modificar las calificaciones se generarán correos especiales exclusivos para los docentes, estos correos serán dados por las personas que tengan correos exclusivos de administradores. Los administradores serán capaces de generar los correos de los estudiantes y de los maestros además que estos podrán modificar los datos de ambos.

# Diagrama relacional de base de datos

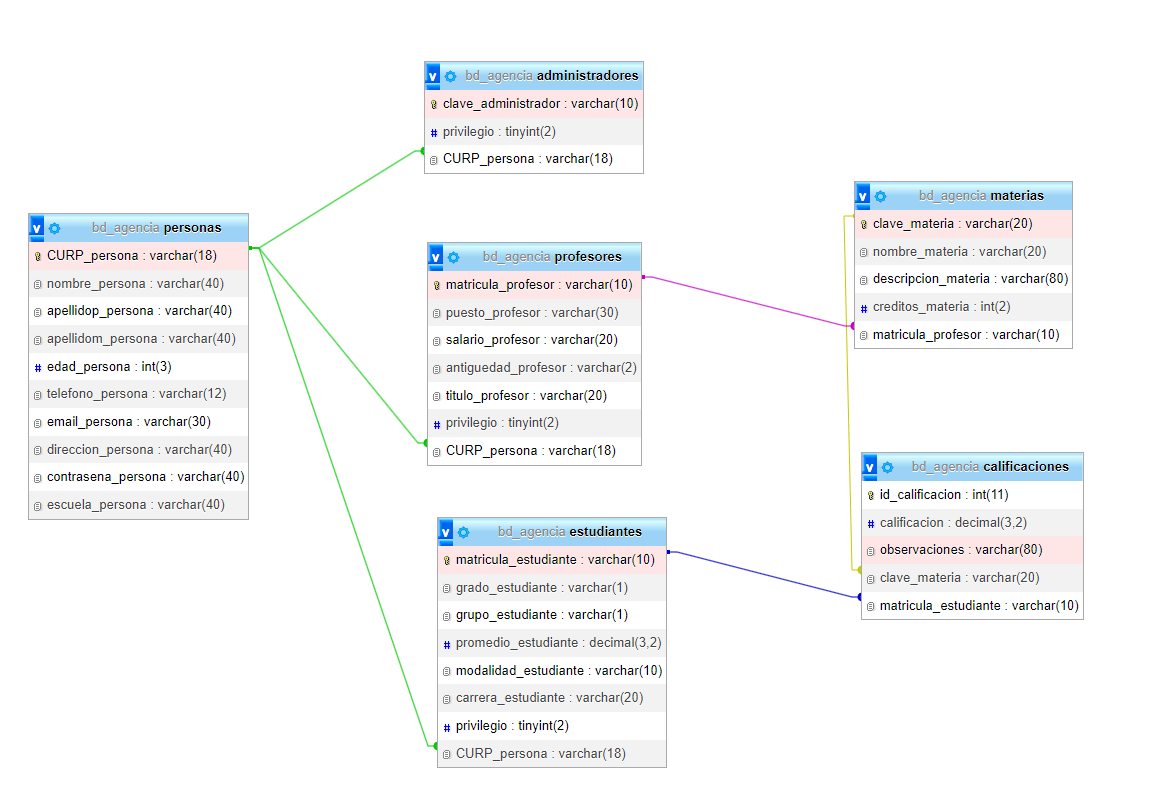


Ilustración Diagrama relacional de base de datos

En la imagen se ven seis clases distintas las cuales componen la base de datos estas son, la clase principal o clase madre la cual tiene como llave primaria Curp\_persona esta se hereda a las tablas administradores, profesores y estudiantes, las cuales también tienen sus propias llaves primarias.

De profesores nace la tabla materias la cual será usada para más adelante, conectar la tabla calificaciones y asi poder guardar las calificaciones subidas de los profesores.

Estudiantes esta directamente relacionada con calificaciones para asi poder mandar las calificaciones a los estudiantes.

# Diagrama UML de clases y objetos

## Ejercicio 1

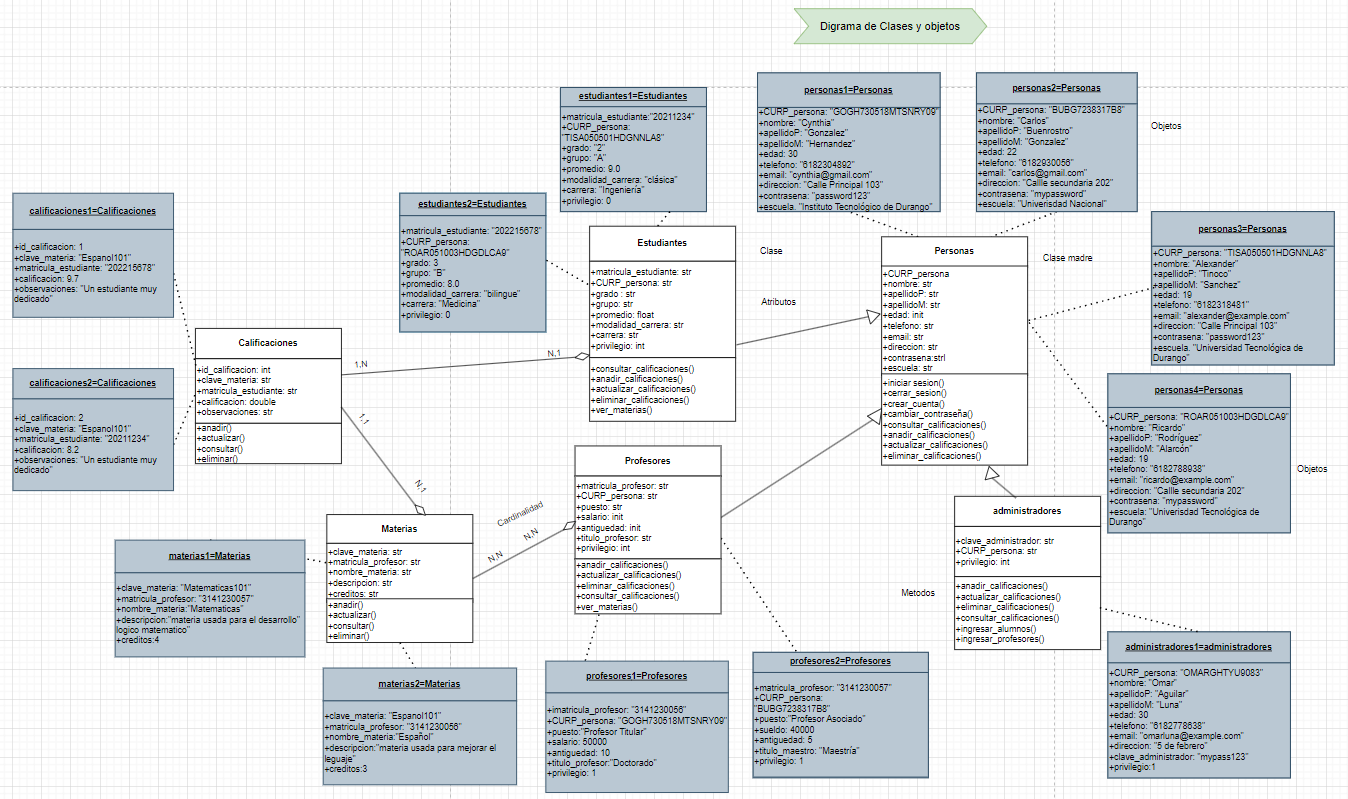


Ilustración Diagrama de objetos y clases

*Descripción del Diagrama de objetos y clases*

El diagrama de clases y objetos muestra nuestro sistema de gestión escolar que organiza la información de las entidades personas, estudiantes, profesores, administradores, materias y calificaciones. La clase `Personas` actúa como la clase madre, de la cuales se heredan las clases `Estudiantes`, `Profesores` y `Administradores`. Los atributos de la clase `Personas` incluyen CURP, nombre, apellidos, edad, teléfono, email, dirección, contraseña y escuela, mientras que los métodos incluyen iniciar y cerrar sesión, cambiar contraseña, consultar y actualizar calificaciones, ver materias.

La clase `Estudiantes` contiene atributos adicionales como matrícula, grado, promedio, modalidad de carrera, carrera y privilegio. La clase `Profesores` incluye atributos como matrícula de profesor, puesto, antigüedad, salario, título y privilegio, junto con métodos para añadir, actualizar y eliminar calificaciones, ver materias y asignar materias. Por su parte, la clase `Administradores` posee un atributo adicional de clave de administrador y métodos que permiten añadir y modificar alumnos y profesores.

La clase `Materias` muestra las asignaturas del sistema, los atributos clave presentados son materia, nombre de materia, matrícula del profesor, descripción y créditos. La clase `Calificaciones` almacena las calificaciones que obtienen los estudiantes en las materias, con sus atributos respectivos ID de calificación, clave de materia, matrícula del estudiante, calificación y observaciones.

*Capturas de pantalla de la codificación conexión*

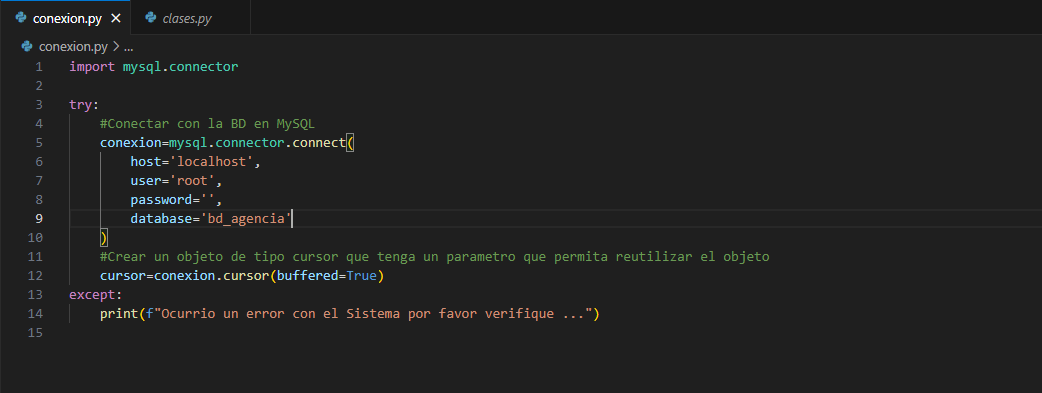


Ilustración 1 conexión a MySQL

La imagen presenta el archivo de “conexión”, el cual crea la conexión a la base de datos también se crea el objeto cursor mediante la conexión este objeto será útil más adelante en el código.

*Capturas de pantalla de la codificación funciones*

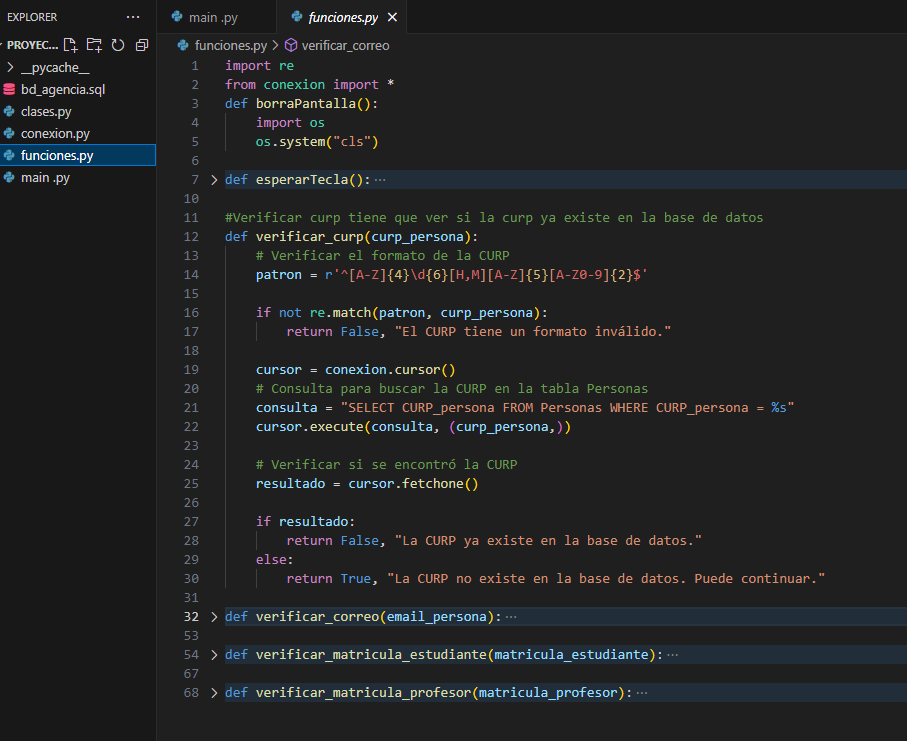


Ilustración 2 funciones

En la ilustración se ve el archivo de funciones donde se presentan todos los métodos usadas para el sistema la mayoría se usaron para generar verificación a la hora de llenar campos y de esta manera si algún usuario llegara a escribir algún campo mal le aparecería una ventana indicándole su error.

*Capturas de pantalla de la codificación clases*

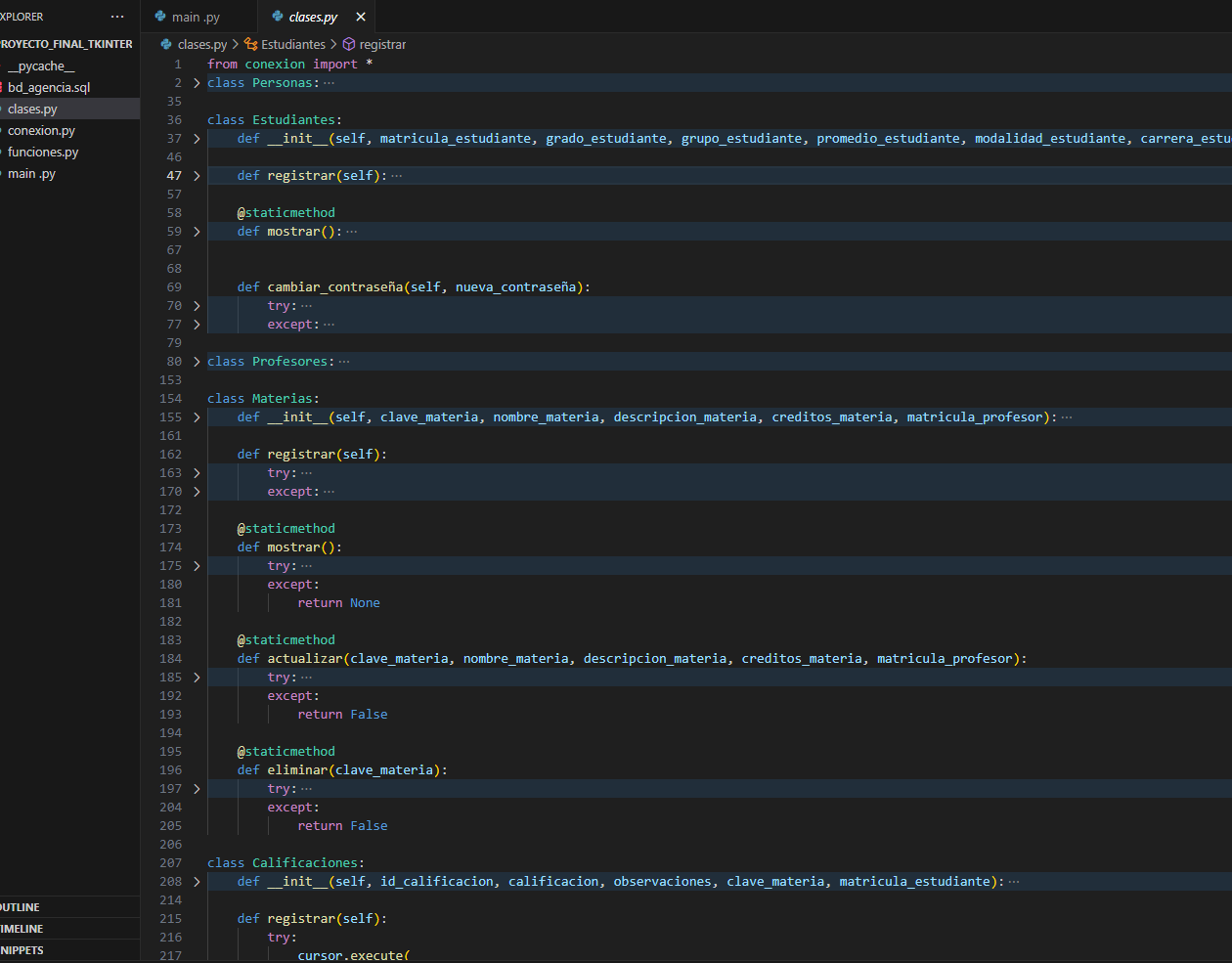


Ilustración 4 clases 1

En la imagen antecesora (siendo la segunda parte del código clases) se ve como se definen nuevas clases y métodos como lo es la clase “Profesores”, los métodos “mostrar()” y “cambiar\_contraseña()”

*Capturas de pantalla de la codificación clases*

Ilustración 3 clases 1

En la imagen anterior se pueden ver las distintas clases usadas por ejemplo personas la cual es la clase principal, estudiantes donde se definen todos sus atributos por medio del constructor (\_init\_) también se ve el método registrar y mostrar más abajo se pueden ver la clase de profesores definiendo nuevos atributos para esta clase. Debajo de esto se ven los métodos para registrar, mostrar, actualizar y eliminar profesores.

*Capturas de pantalla de la codificación clases*

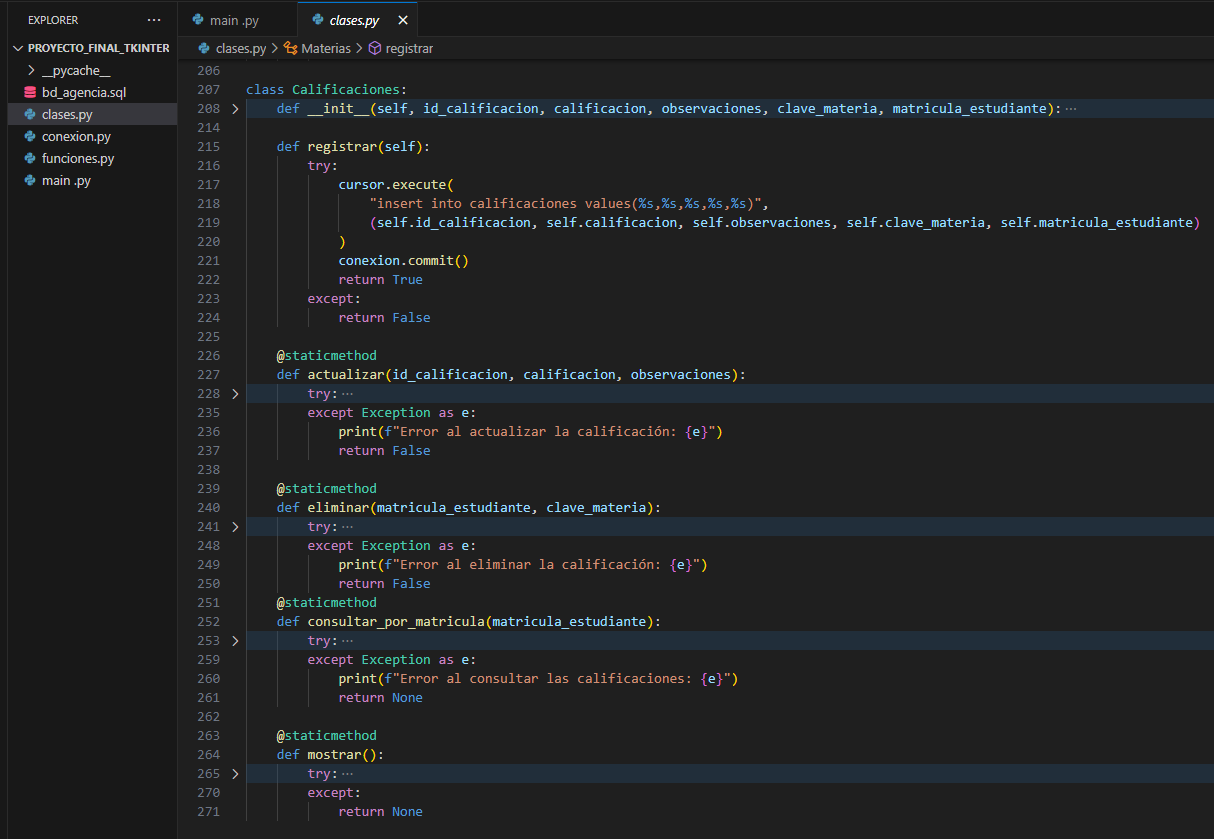


Ilustración 4 clases 2

En esta captura se ve la creación de la clase calificaciones donde además también se puede notar la creación de los métodos para mostrar, registrar, actualizar o eliminar las calificaciones. En la parte final se observa la creación de métodos para consultar las calificaciones por medio de la matrícula y un método para mostrar las calificaciones, la mayoría de estos usando verificación por medio del manejo de excepciones “try” y “except”

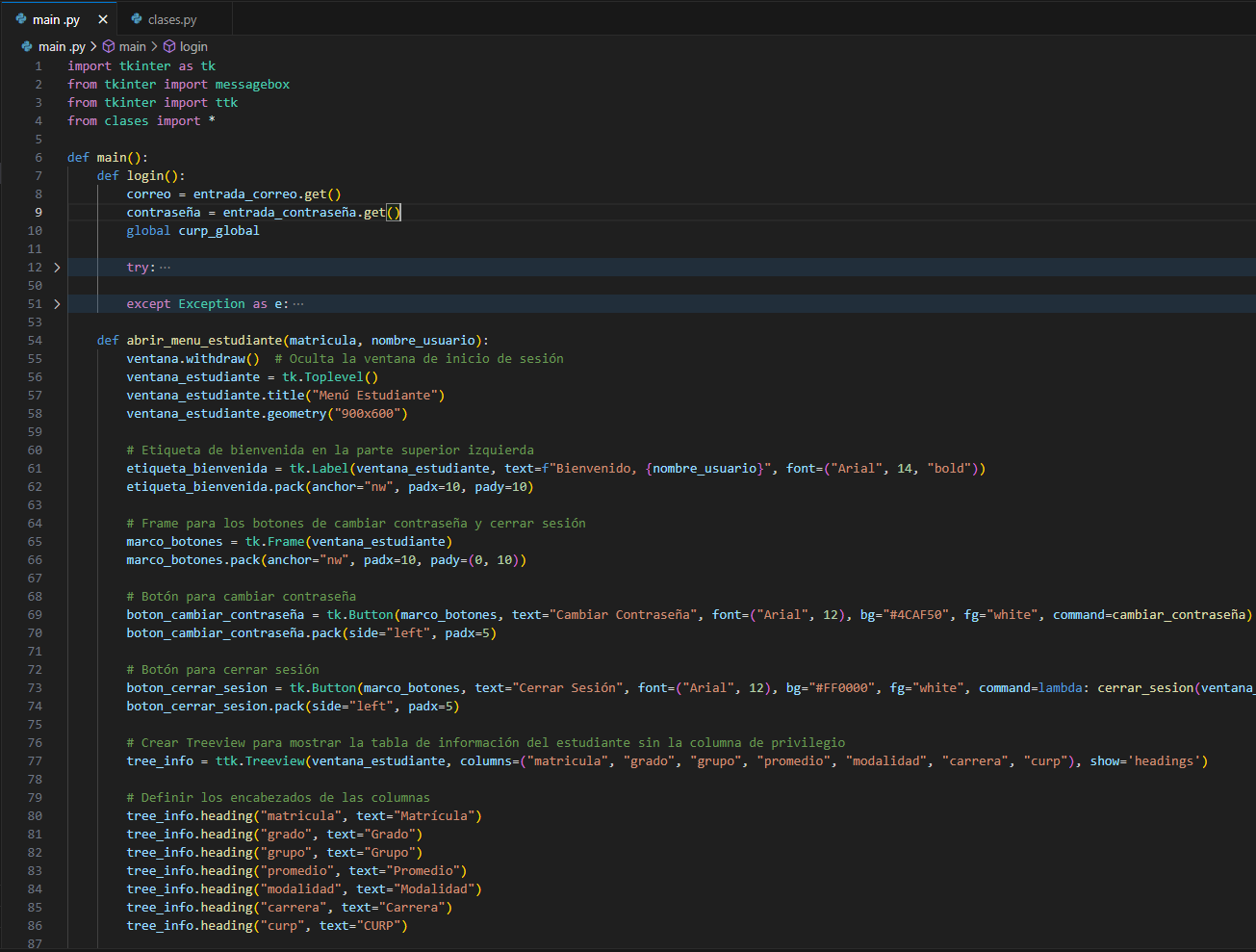
*Capturas de pantalla de la codificación main*

Ilustración 5 login y menú estudiante

En la siguiente ilustración se ve como se importa tkinter para crear las interfaces además de importar también las cajas de mensaje esta usadas para mandar un mensaje cuando se ejecute nuestra verificación un poco mas abajo encontraremos la creación del método login usado para ingresar a la ventana principal y debajo de este se ve la creación de la ventana de estudiantes a la cual solo se puede entrar si tienes un correo exclusivo de estudiante.

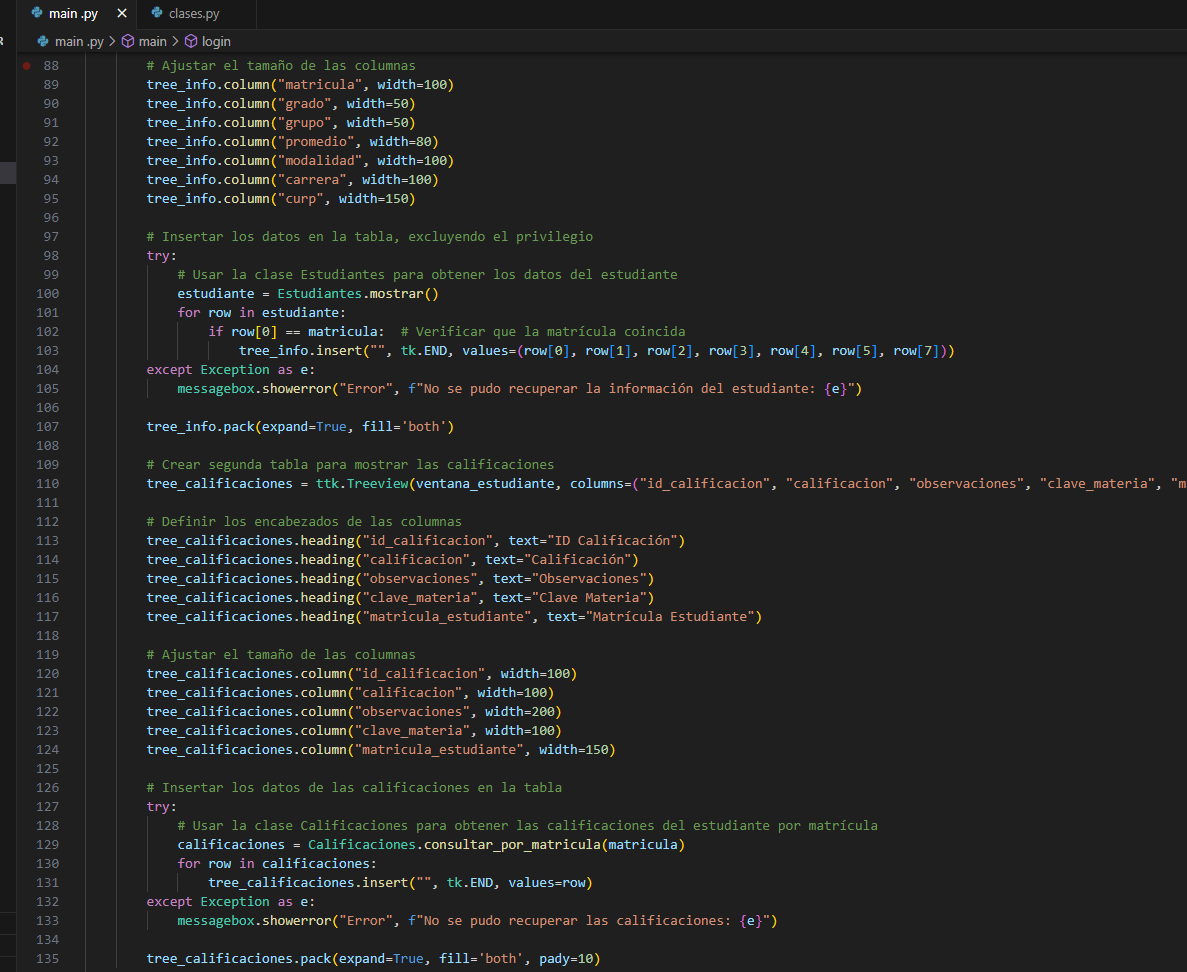
*Capturas de pantalla de la codificación main*

Ilustración 6 menú estudiantes 2

En la captura se pueden ver las distintas instrucciones generadas para darle diseño a la ventana de estudiantes además de las instrucciones para insertar los datos de los estudiantes e insertar sus calificaciones.

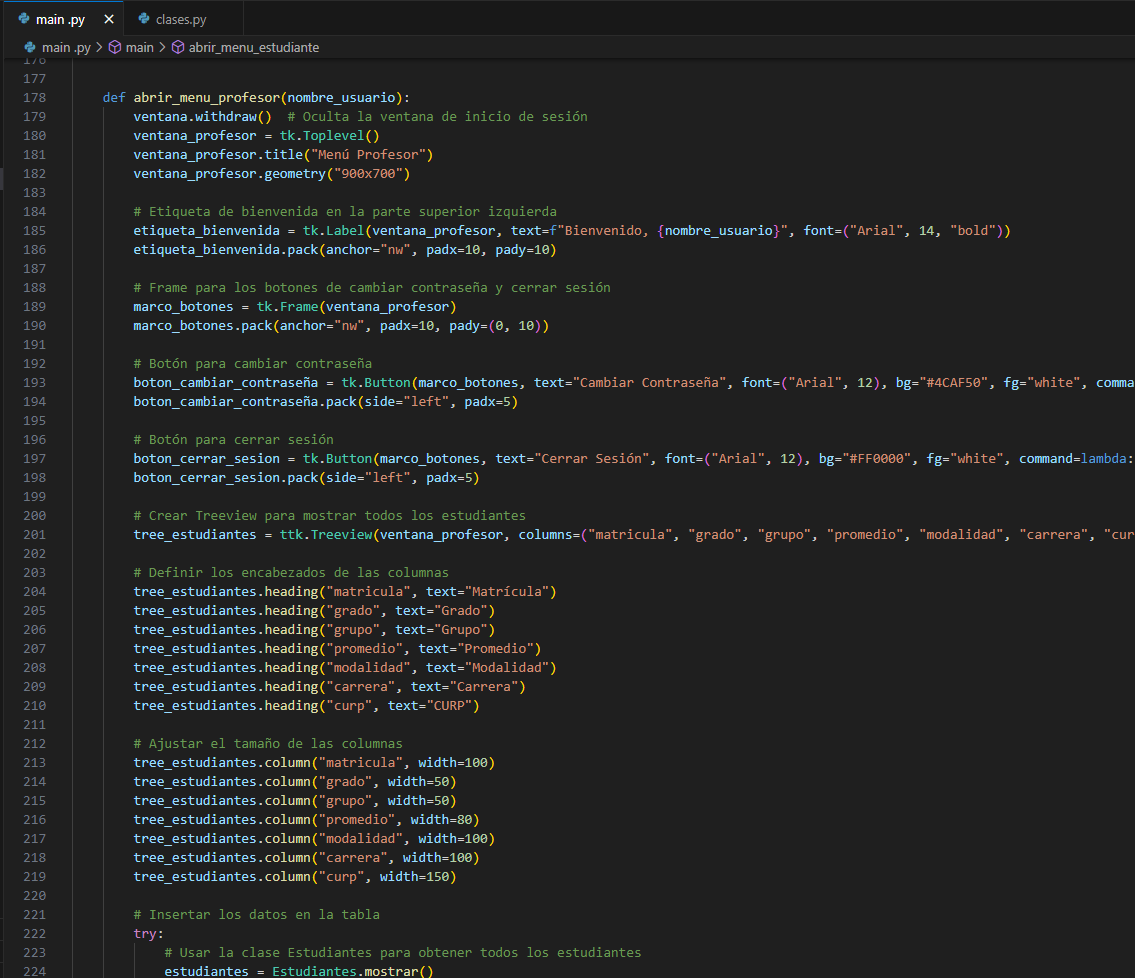
*Capturas de pantalla de la codificación main*

Ilustración 7 ventana profesores

En la imagen se ve la codificación para la generación de la ventana de profesores la cual solo debería de aparecer si el correo y la contraseña son de un profesor que ya está dado de alta a la base de datos. También se hace uso de varias instrucciones para agregar distintas cosas a la ventana como los botones para cambiar contraseña o cerrar sesión o definir o ajustar el tamaño de las columnas.

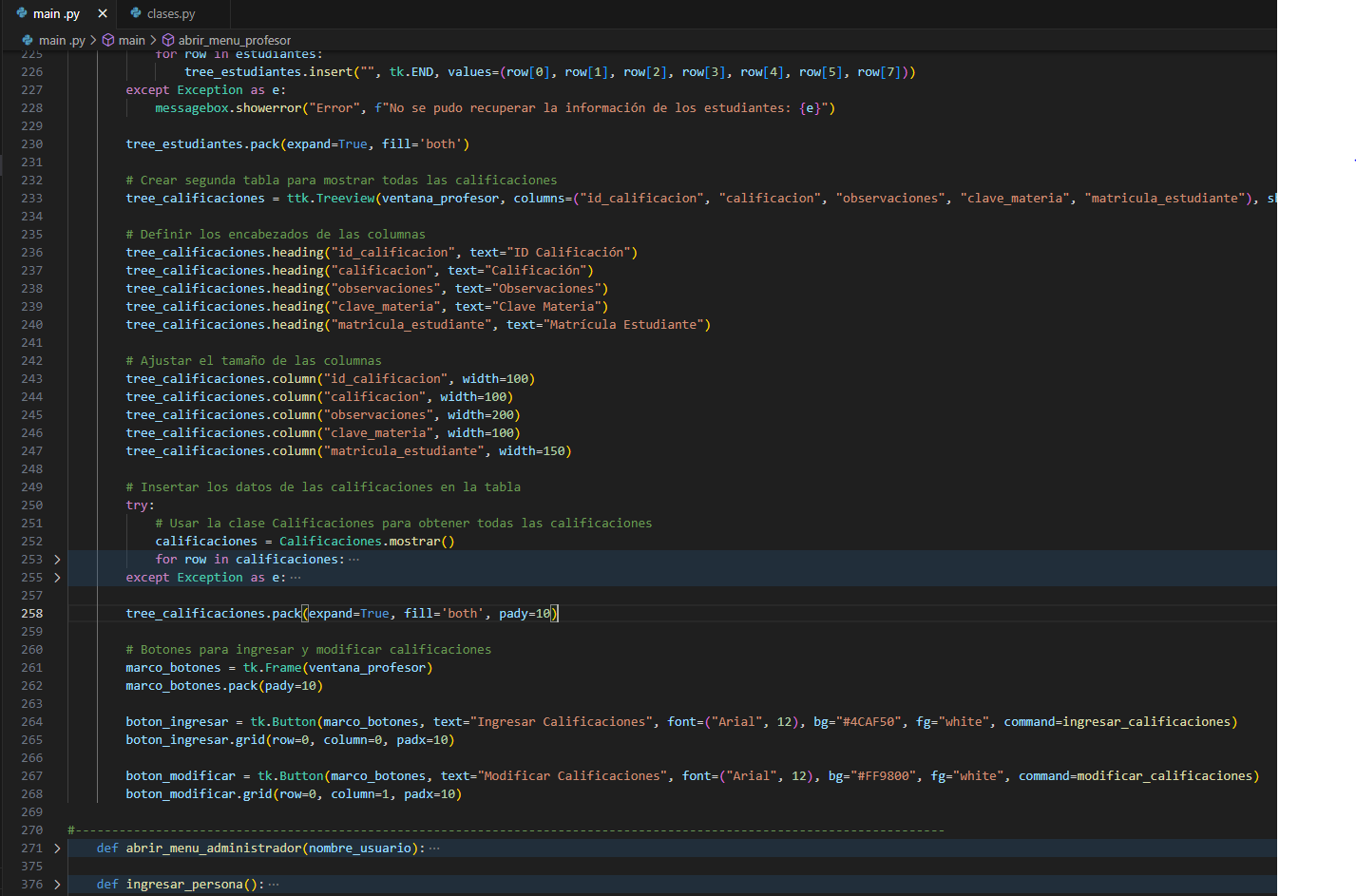
*Capturas de pantalla de la codificación main*

Ilustración 8 insertar calificaciones

En la ilustración se pueden ver componentes de la biblioteca tkinter, como Treeview y Button, y maneja errores potenciales mediante bloques try-except. El uso principal que se le da a Treeview en la imagen es la de mostrar las calificaciones en la ventana de profesores.

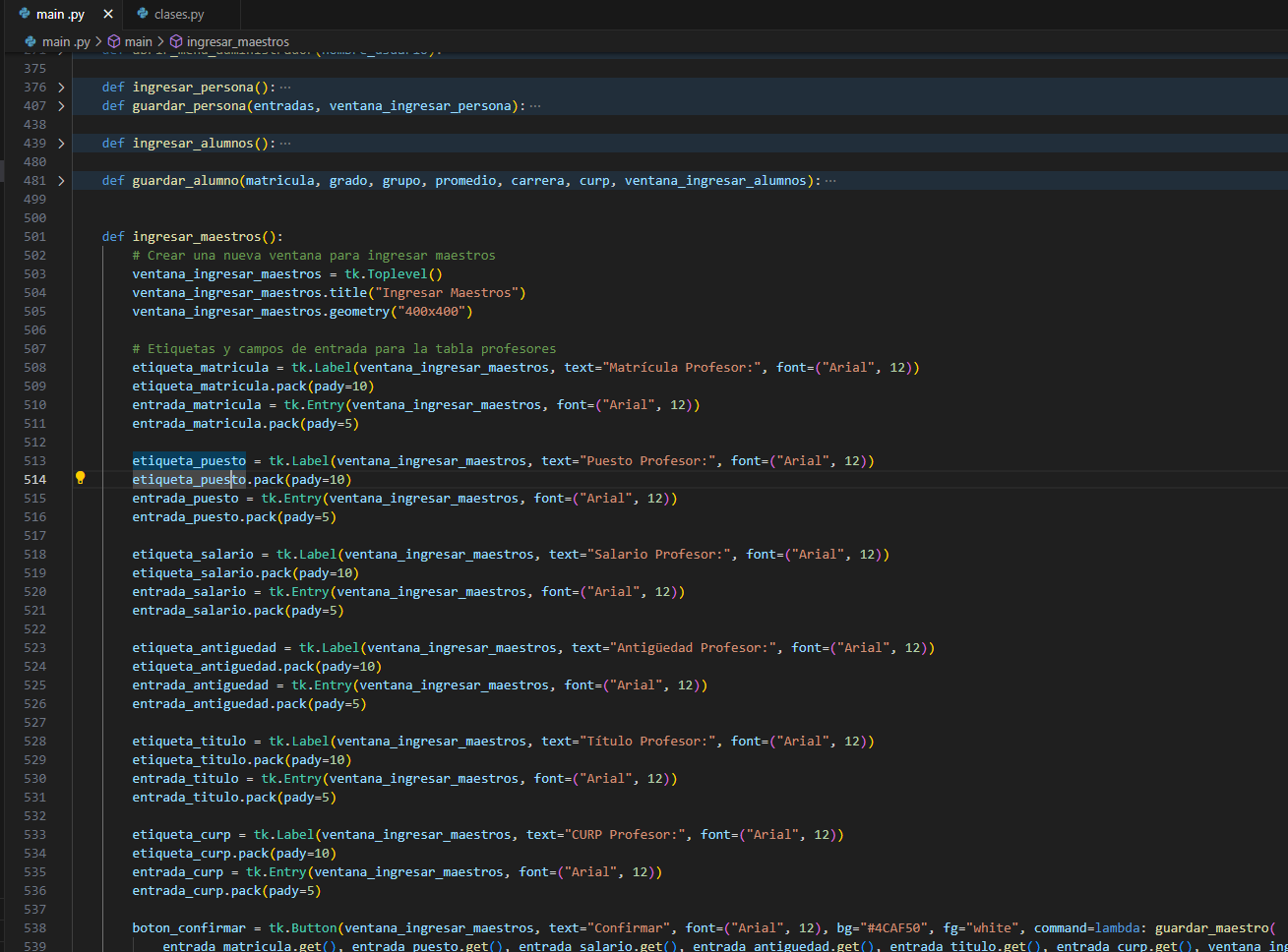
*Capturas de pantalla de la codificación main*

Ilustración 9 agregar maestros y personas

En la ilustración 8 se ve la codificación para la ventana para ingresar los datos de los maestros a la hora de agregarlo a la base de datos también se ve la creación de la ventana para agregar personas a la base datos tales como estudiantes o maestros y por último la creación de la ventana para agregar alumnos.

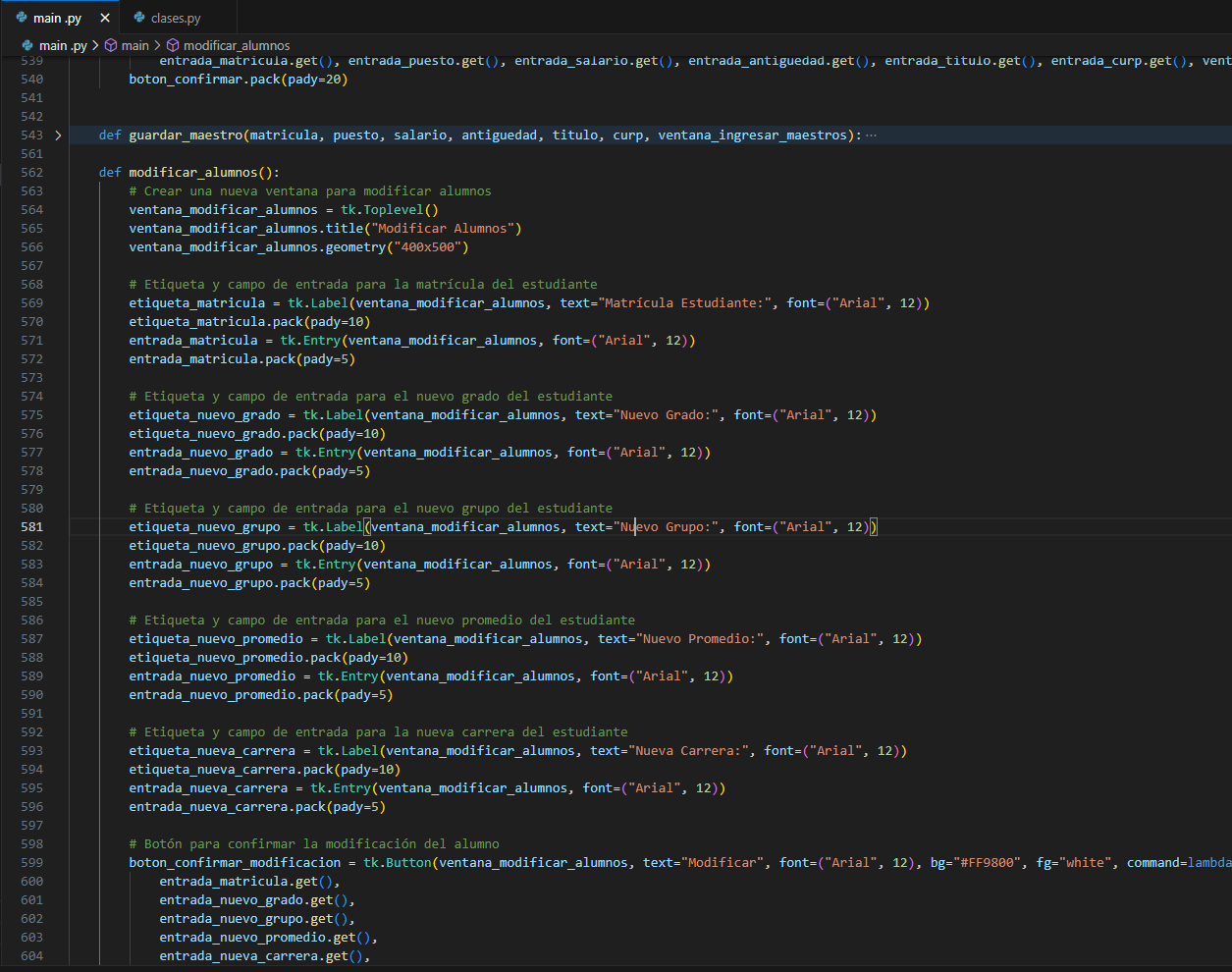
*Capturas de pantalla de la codificación main*

Ilustración 10 main modificar alumnos

En gran parte de la captura se puede ver como se crea la ventana para modificar a los alumnos usando etiquetas y entradas (tkt.label y tkt.entry) todas estas conformar el código para poner campos donde irán los datos de los alumnos al cual se desea modificar.

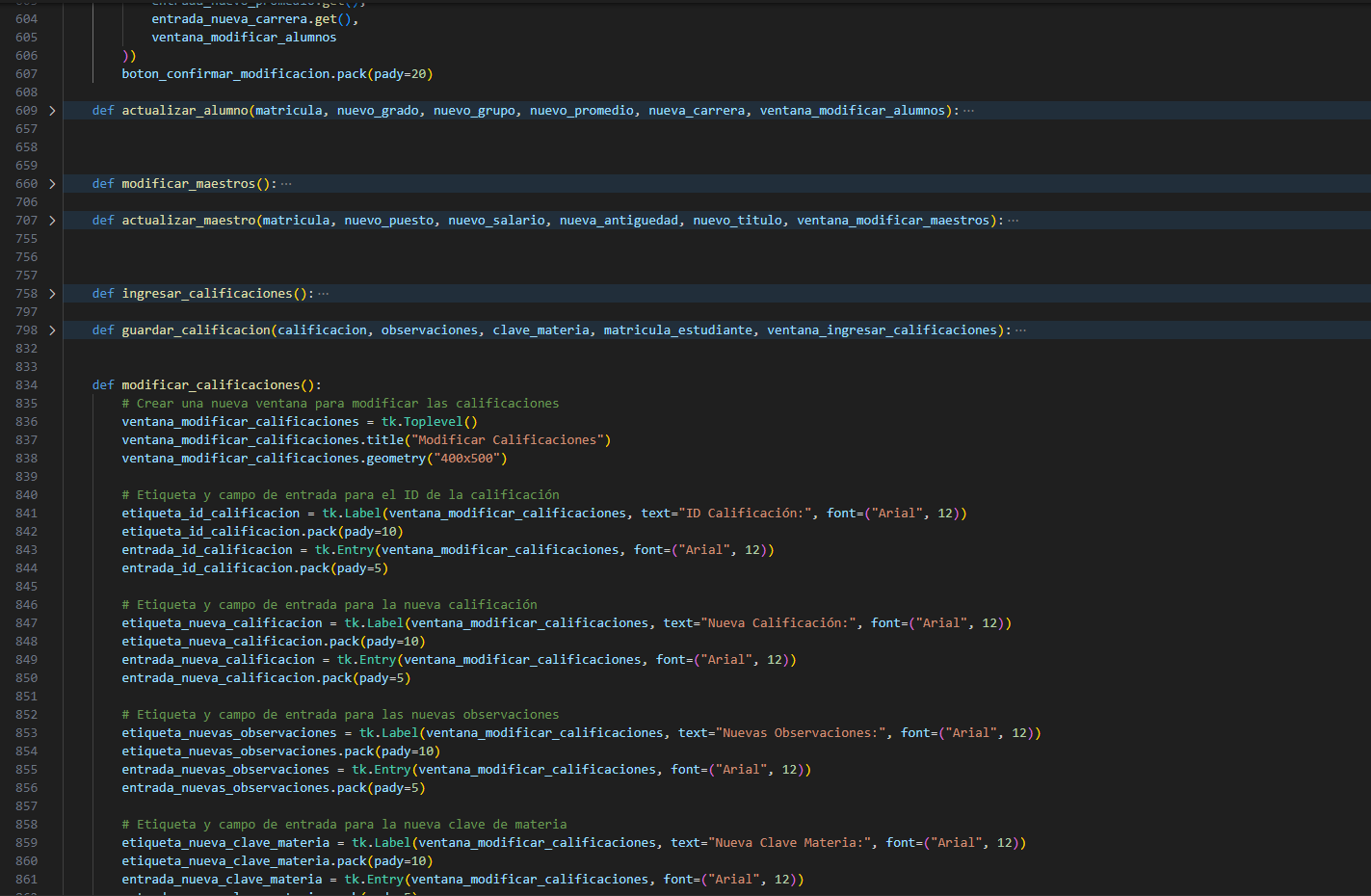
*Capturas de pantalla de la codificación main*

Ilustración 11 modificar calificaciones

En la ilustración 10 se ve la creación de distintos métodos para realizar muchas acciones dentro de las cuales están modificar maestros, actualizar maestros, actualizar alumnos y modificar calificaciones todos estos métodos están creados para generar ventanas en las cuales se puedan escribir los nuevos datos que se desean actualizar o modificar.

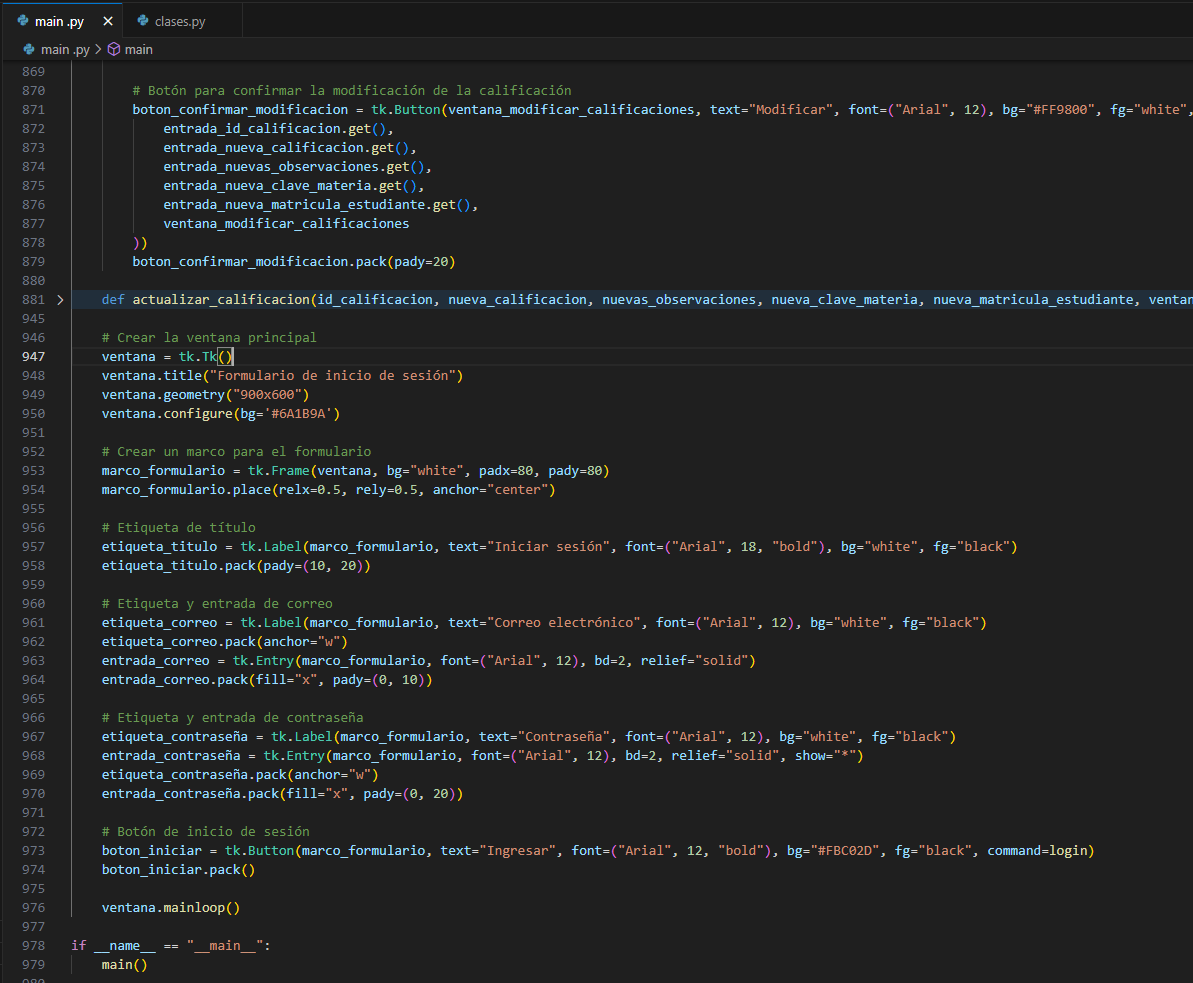
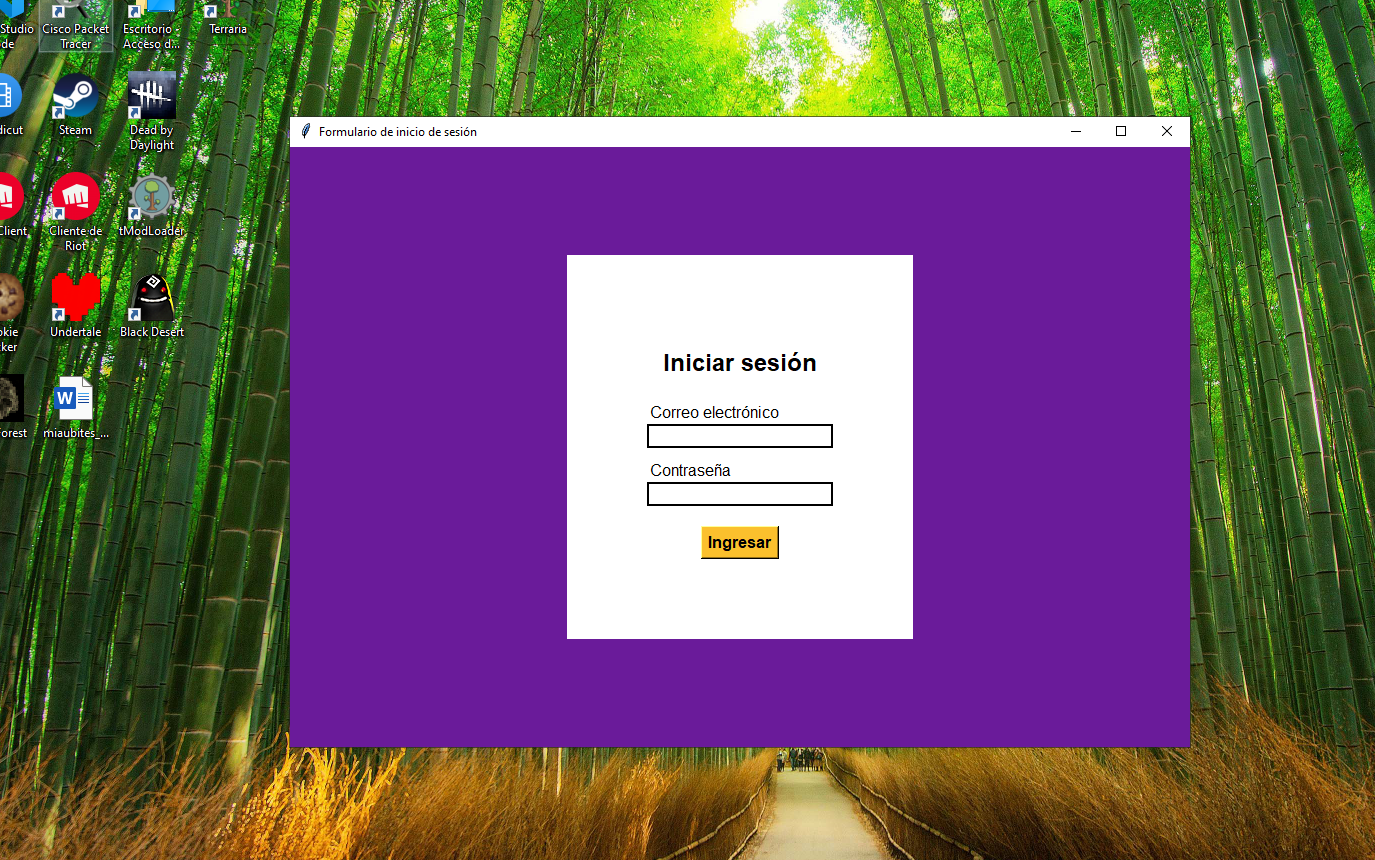
*Capturas de pantalla de la codificación main*

Ilustración 12 actualizar calificaciones

En la última imagen del main se ve la creación del botón para la modificación de calificaciones y en la parte de abajo se ve la creación de un método para abrir una ventana para actualizar las calificaciones aún más abajo se puede ver la creación del menú principal donde se pedirá el correo y una contraseña para poder ingresar al sistema.

*Capturas de pantalla de la ejecución*

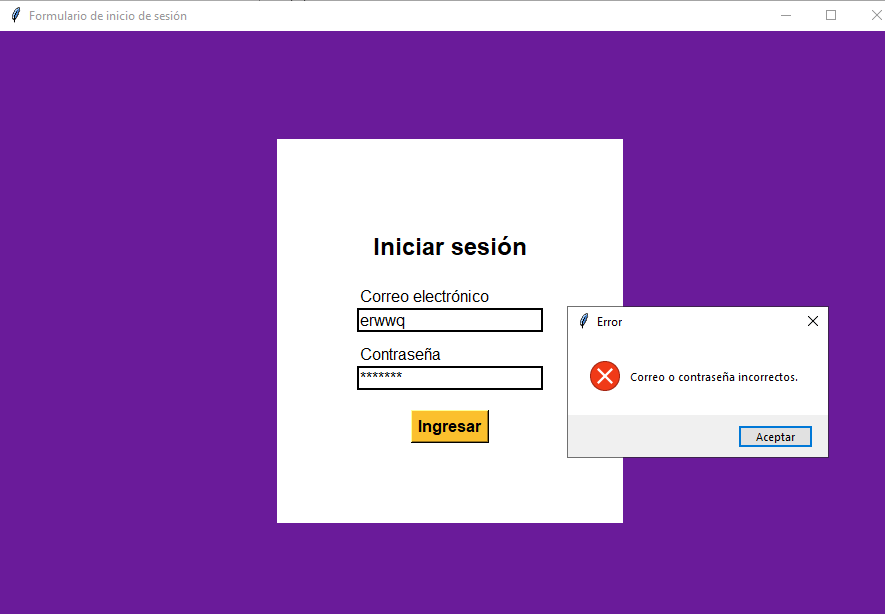
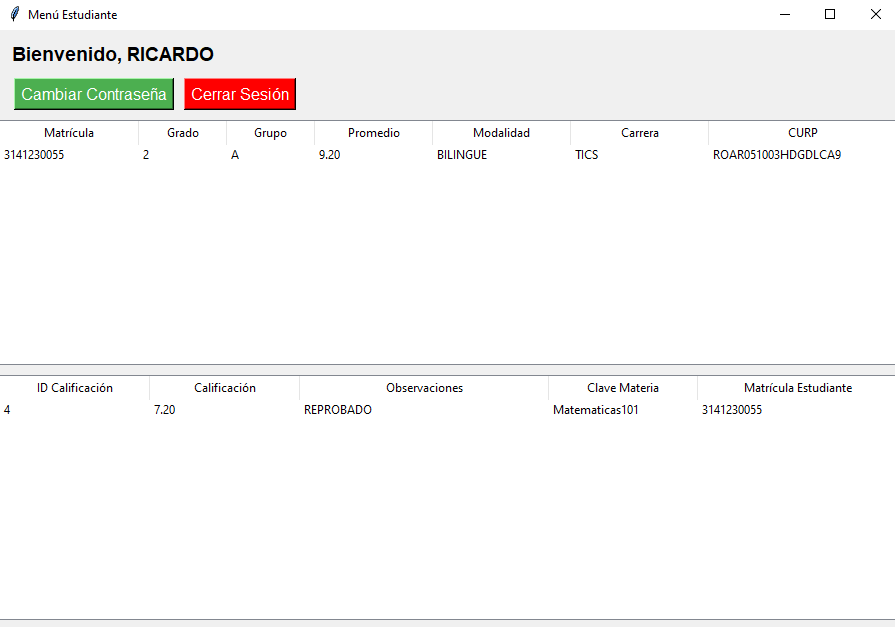


Ilustración 13 ejecución 1

En las capturas se ve la pestaña de inicio de sesión de nuestra aplicación de escritorio donde podremos poner nuestro correo electrónico y contraseña para acceder al sistema. Si escribimos un correo o contraseña erróneo un mensaje aparecerá para indicarnos que usemos un correo o contraseña que estén registrados

*Capturas de pantalla de la ejecución*



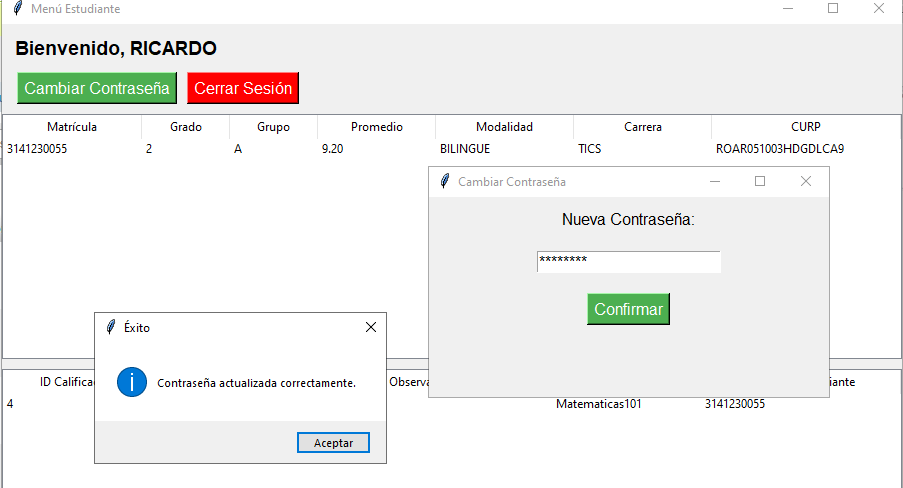


Ilustración 14 ejecución 2

En la imagen se ve como si ponemos un correo para estudiantes se generará otra ventana donde se verán los datos del alumno y las calificaciones del mismo. También habrá dos botones uno para cerrar sesión y otro para cambiar la contraseña si cambiamos la contraseña aparecerá un mensaje de que nuestra contraseña ha sido cambiada.

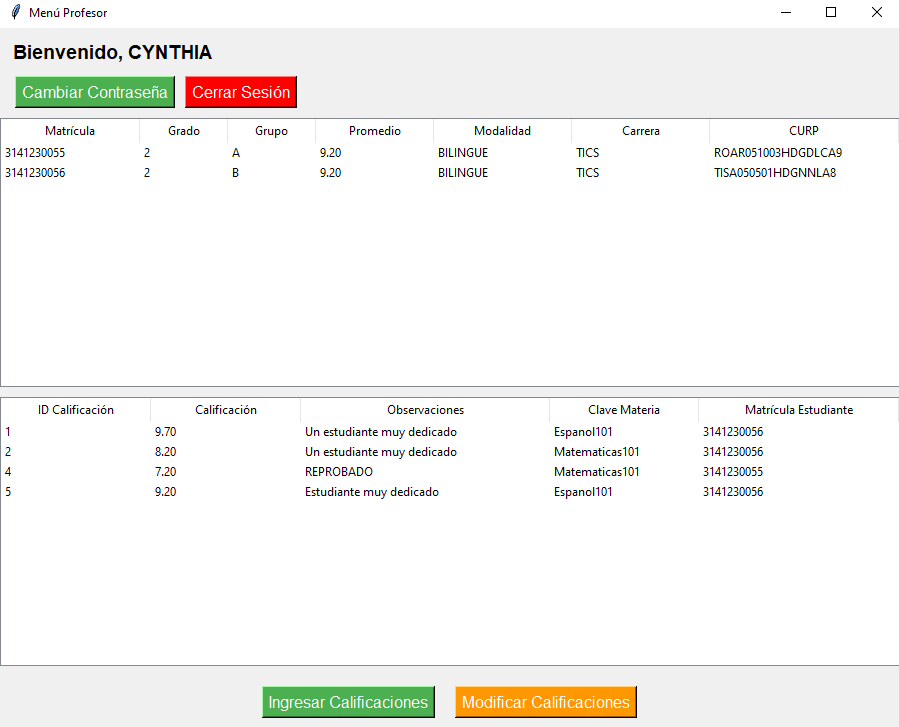
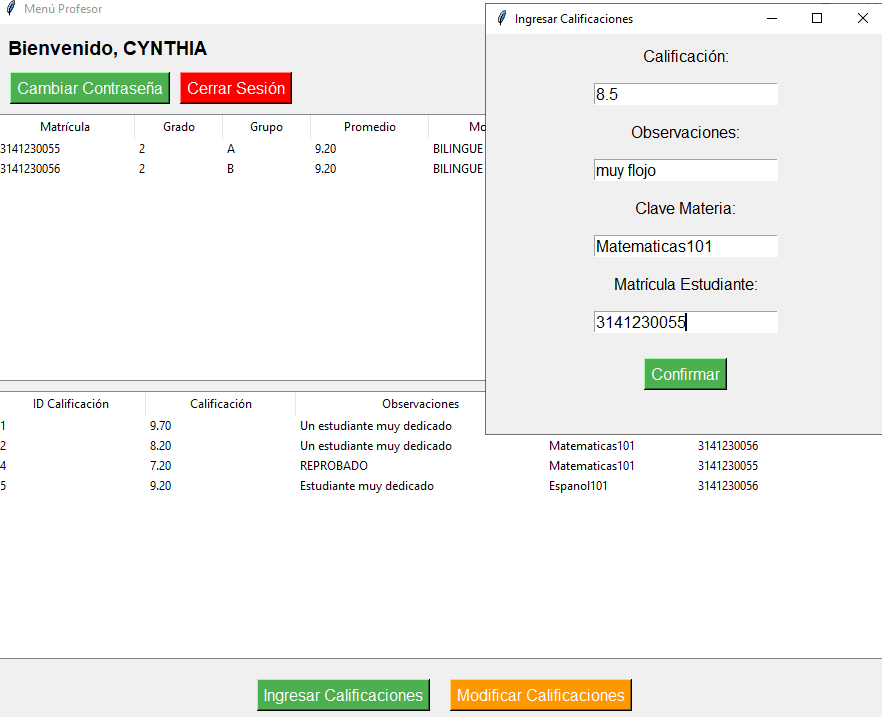
*Capturas de pantalla de la ejecución*

Ilustración 15 ejecución 3

En las imágenes se ve la ventana al poner un correo de maestro en este también se puede cambiar o cerrar sesión la diferencia con una cuenta para alumnos se podrá agregar calificaciones o modificarlas

*Capturas de pantalla de la ejecución*



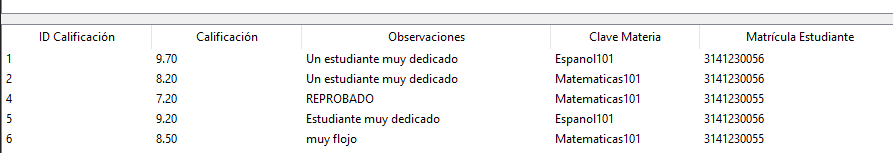


Ilustración 16 ejecución 4

Las capturas muestran como al ingresar la calificación nos pedirá distintos campos y al llenarlos toda la nueva calificación será subida. Para que los cambios se vean en pantalla se necesitara cerrar la sesión y volver a entrar nuevamente.

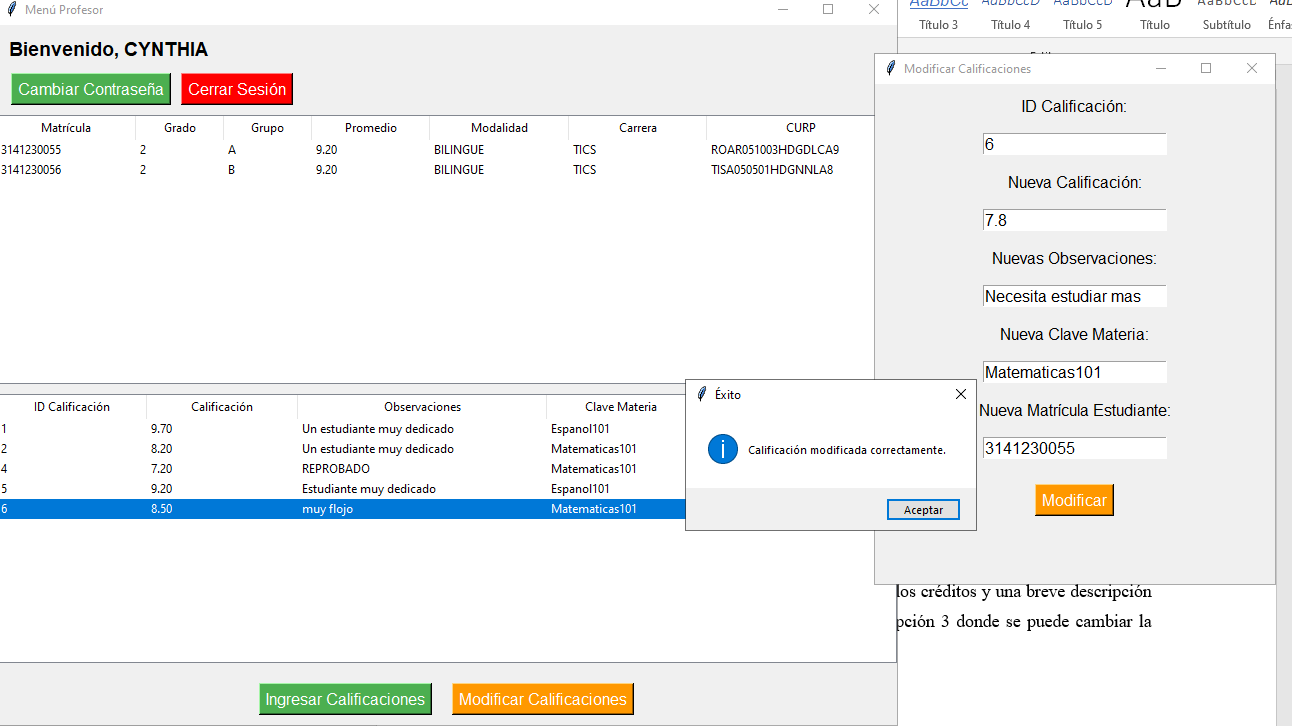
*Capturas de pantalla de la ejecución*



Ilustración 17 ejecución 5

Las capturas muestran como al presionar el botón de modificar calificaciones se generará otra ventana donde deberemos colocar todos los campos al finalizar se subirán nuestros cambios. Para esto debemos cerrar sesión y entrar de nuevo.

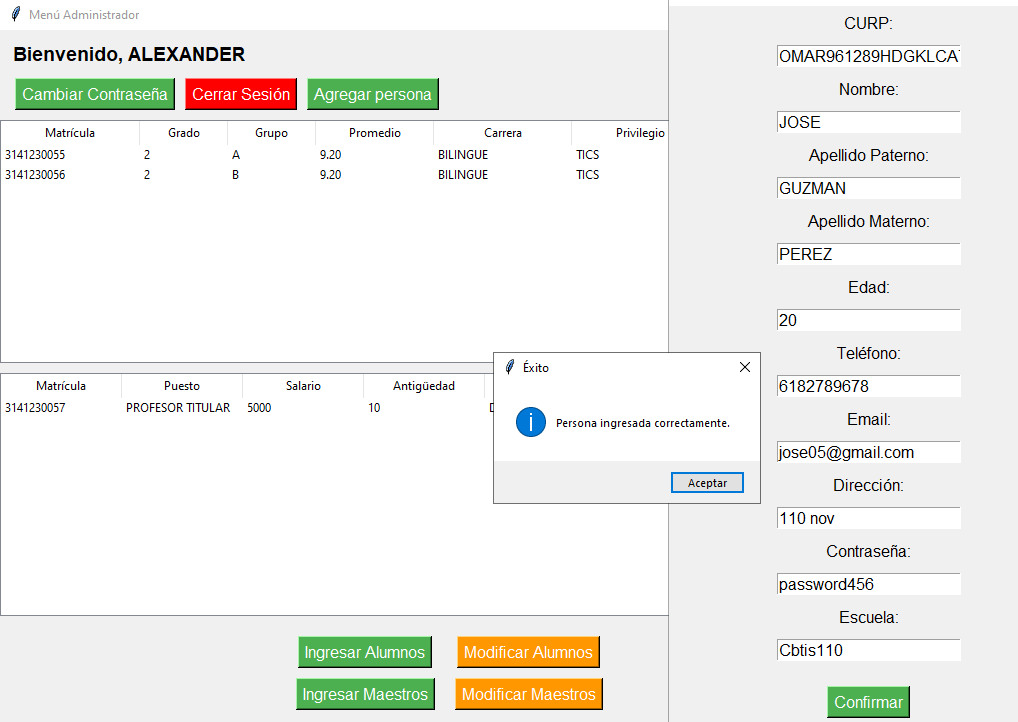
*Capturas de pantalla de la ejecución*

Ilustración 18 ejecución 6

La imagen muestra como al ingresar como administrador con un correo exclusivo tendremos nuevas opciones como agregar personas, agregar alumnos, agregar maestros o modificar y modificar maestros. Al seleccionar la opción de agregar personas podremos después agregarlos como alumnos o maestros.

## Capturas de pantalla de la ejecución

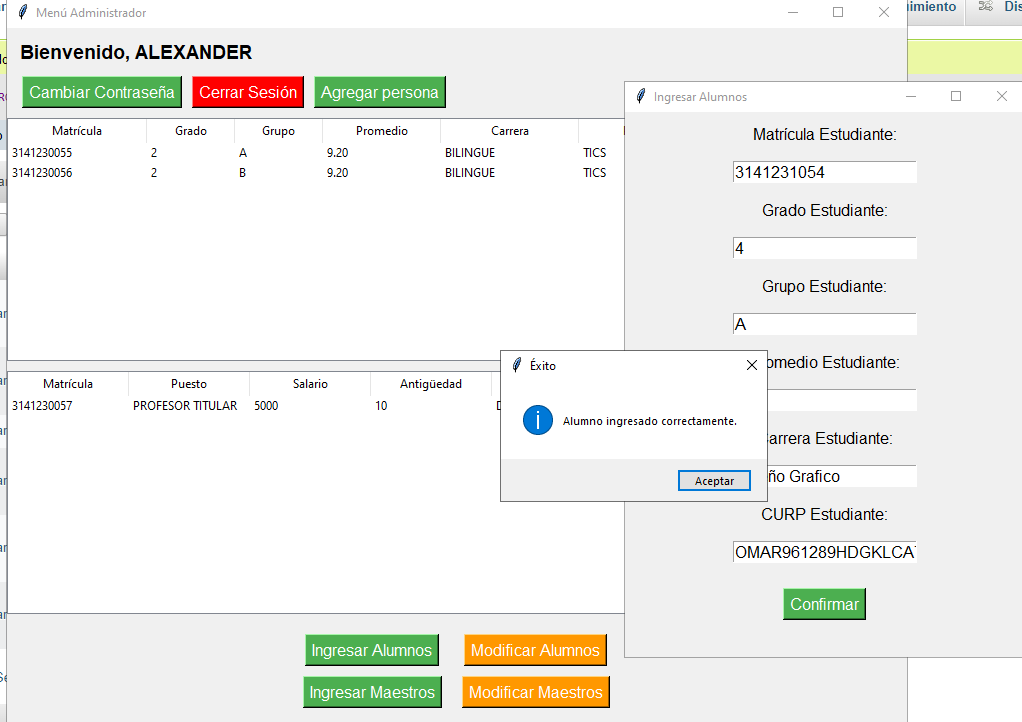




Ilustración 19 ejecución 7

En la imagen se ve como al seleccionar la opción de alumnos podemos agregarlos con la Curp que creamos al agregar una persona. Para ver estos cambios tendremos que cerrar sesión y volver a iniciar.

## Capturas de pantalla de la ejecución

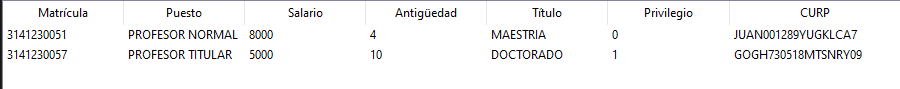


Ilustración 20 ejecución 8

En la ilustración se muestra lo que ocurre al presionar el botón de ingresar un maestro. Al hacer esto podremos ingresar los datos del nuevo maestro ya antes agregado en personas. Para que los datos sean registrarlos se deberá cerrar y abrir sesión.

## Capturas de pantalla de la ejecución

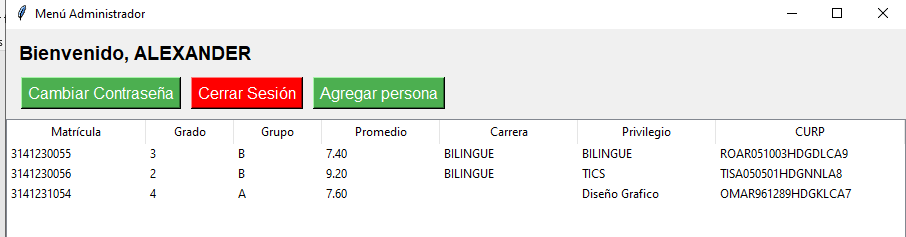
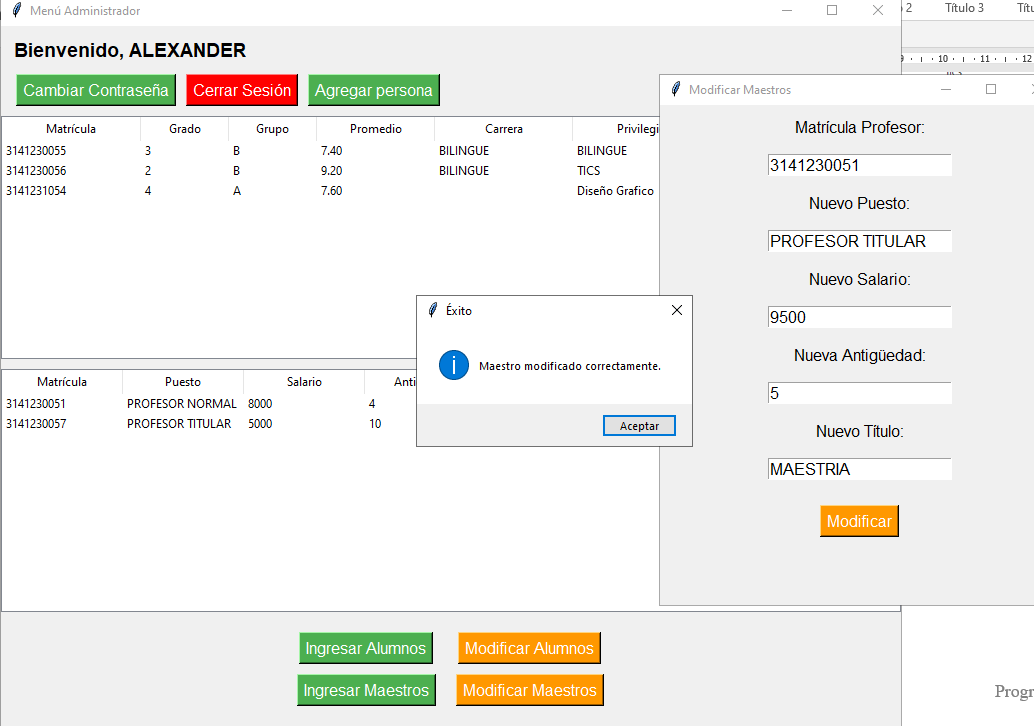


Ilustración 21 ejecución 9

En las ilustraciones se muestra lo que pasa al presionar el botón modificar alumno en donde se podrían cambiar distintos campos como el grado, grupo por si este ha pasado de grado o ha sido cambiado de grupo.

## Capturas de pantalla de la ejecución



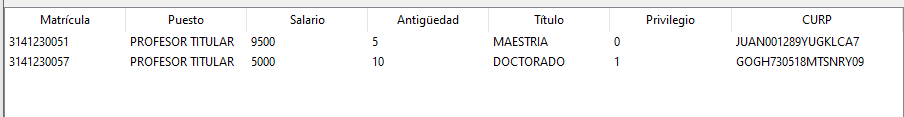


Ilustración 22 ejecución 10

En las ilustraciones se ve que pasa cuando eliges la opción de modificar un maestro te pedirá ciertos campos y al cerrar sesión y volver a iniciar sesión los datos se habrán guardado.

# Conclusiones

Durante este cuatrimestre, yo y Alexander, hemos logrado un progreso significativo en nuestro aprendizaje sobre Python. Comenzamos sin tener mucho conocimiento sobre el lenguaje, enfrentándonos a conceptos completamente nuevos como la programación orientada a objetos, la creación de métodos, clases, ciclos y manejo de excepciones. A lo largo del tiempo, fuimos avanzando paso a paso, aprendiendo a conectar Python con MySQL para manejar bases de datos y luego, incluso, a desarrollar interfaces gráficas con `tkinter`.

Este viaje de aprendizaje finalizo en nuestro proyecto final: un sistema de gestión escolar. Este proyecto no solo consolidó todo lo que habíamos aprendido, sino que también nos permitió aplicar esos conocimientos en un contexto real y útil, creando una herramienta que podría ser de gran utilidad para el Centro de Bachillerato Tecnológico Industrial y de Servicios Número 110.

Mirando hacia atrás, es impresionante ver cómo hemos pasado de casi no saber nada de Python a desarrollar un sistema completo que integra múltiples aspectos de la programación. Este cuatrimestre no solo nos enseñó habilidades técnicas, sino que también fortaleció nuestra capacidad para resolver problemas y trabajar en equipo. Sin duda, los logros alcanzados reflejan el esfuerzo y la dedicación que hemos puesto en este proyecto.

Ricardo Rodríguez Alarcón

Alexander Tinoco Sánchez

# Bibliografía

* Creator’s Academy. (2023, May 23). Aplicaciones de escritorio en Python: Desarrollo de interfaces gráficas con Tkinter [Video]. YouTube. <https://www.youtube.com/watch?v=7t65ss0amnw>
* Programación y más. (2023, 5 de julio). Cómo hacer interfaces gráficas con Tkinter en Python [Video]. YouTube. <https://www.youtube.com/watch?v=MpkTYMzhV0A&t=812s>
* Wikibooks. (s.f.). Python/Interfaz gráfica con Tkinter/Gestión de ventanas. Wikilibros. Recuperado el 17 de agosto de 2024, de <https://es.wikibooks.org/wiki/Python/Interfaz_gr%C3%A1fica_con_Tkinter/Gesti%C3%B3n_de_ventanas>
* Recursos Python. (s.f.). Botón Button en Tkinter. Recursos Python. Recuperado el 17 de agosto de 2024, de [Project Timeline](https://recursospython.com/guias-y-manuales/boton-button-en-tkinter/) <https://recursospython.com/guias-y-manuales/boton-button-en-tkinter/>
* PythonYa. (s.f.). Funciones en Python. Tutoriales Programación Ya. Recuperado el 16 de agosto de 2024, de [Our Services](https://tutorialesprogramacionya.com/pythonya/detalleconcepto.php?codigo=57) <https://tutorialesprogramacionya.com/pythonya/detalleconcepto.php?codigo=57>
* Stack Overflow en español. (2018, octubre 1). ¿Se pueden crear llaves foráneas desde phpMyAdmin MySQL? Stack Overflow en español. <https://es.stackoverflow.com/questions/201982/se-pueden-crear-llaves-for%C3%A1neas-desde-phpmyadmin-mysql>