

Documentació proyecto final programación IV

Integrantes del grupo:

-Daniel Rico Palacio

-Amir Evelio Hurtado Mena

Contexto del proyecto:

El contexto o la temática del proyecto se basa en la gestión de una institución a pequeña escala, permitiendo en él la gestión de profesores, la gestión de estudiantes y sus notas en las diferentes materias en las cuales estén matriculados.

flujo de trabajo:

El proyecto se inició primero con la planeación del proyecto y como cada integrante del grupo iba a desarrollar sus tareas para la culminación de final del proyecto, como primera tarea fue la abstracción de las diferentes clases, métodos y funciones usadas en el código, la segunda tarea fue armar todo el código en uno solo uniendo cada parte para que funcionara e interactúa correctamente con las otras partes, como tercera tarea fue la realizar las soluciones a los diferentes bugs y errores que aparecían a la hora de ejecutar el código, habiendo diferentes y algunos errores similares en las primeras interfaces dejando una enseñanza de como diseñar las nuevas interfaces, la penúltima tarea fue la modularización de todo el proyecto para mantener un mejor orden y fuera más flexible a la hora de resolver posibles bugs o realizar mejoras a este, por última tarea fue corregir los últimos bugs, errores, cambiar las interfaces para que fueran más atractivas y finalmente terminar la documentación respectiva con el proyecto.

Librerías Implementadas:

- Tkinter

Módulos:

- JSON

Tarjetas CRC:

Documentació projecto final programación IV

Usuario	
RESPONSABILIDADES:	COLABORADORES:
Nombre	Estudiante
Apellido	Profesor
Email	
mostrar_datos()	

Estudiante	
RESPONSABILIDADES:	COLABORADORES:
Nombre	Usuario
Apellido	GestorEstudiantes
Email	BaseDeDatos
Código	
Notas	
Promedio	
mostrar_datos()	

Profesor	
RESPONSABILIDADES:	COLABORADORES:
Nombre	Usuario
Apellido	GestorProfesores
Email	BaseDeDatos
Especialidad	
mostrar_datos()	

Documentació projecto final programación IV

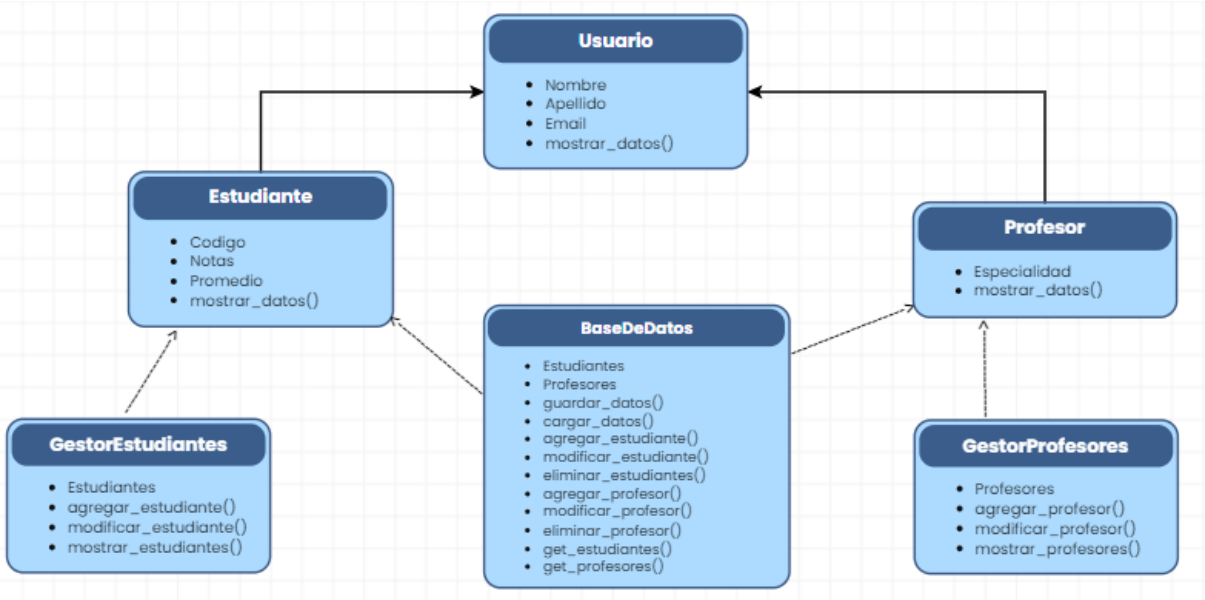
GestorEstudiantes	
RESPONSABILIDADES:	COLABORADORES:
Estudiantes	Usuario
agregar_estudiante()	Estudiante
modificar_estudiante()	
mostrar_estudiantes()	

GestorProfesores	
RESPONSABILIDADES:	COLABORADORES:
Profesores	Usuario
agregar_profesor()	Profesor
modificar_profesor()	
mostrar_profesores()	

Documentació projecto final programación IV

BaseDeDatos	
RESPONSABILIDADES:	COLABORADORES:
Estudiantes	Usuario
Profesores	Estudiante
guardar_datos()	Profesor
cargar_datos()	
agregar_estudiante()	
modificar_estudiante()	
eliminar_estudiante()	
agregar_profesor()	
modificar_profesor()	
eliminar_profesor()	
get_estudiantes()	
get_profesores()	

Diagrama de Clases:



Documentació proyecto final programación IV

Requisitos no funcionales:

Usabilidad:

- La interfaz de usuario debe ser intuitiva y fácil de usar para estudiantes, profesores y administradores.

Rendimiento:

- El sistema debe ser capaz de manejar un número específico de usuarios simultáneos, garantizando un rendimiento aceptable.
- Las consultas de notas y generación de informes deben realizarse de manera eficiente

Mantenibilidad:

- Deben ser posibles las actualizaciones del sistema para agregar nuevas funcionalidades o corregir errores sin afectar su operación normal.

Interoperabilidad:

- Debe seguir estándares aceptados para la transferencia de datos educativos si es aplicable.

Requisitos funcionales:

Gestión de usuarios:

- El sistema debe permitir la creación de cuentas de usuario para estudiantes y profesores.

Registro de notas:

- Los profesores deben poder ingresar y actualizar las notas de los estudiantes.
- El sistema debe permitir el registro de notas por asignatura.

Visibilidad de registros:

- El programa debe permitir la visualización de los estudiantes y profesores registrados

Documentació proyecto final programación IV

Casos de uso:

Mostrar Datos del Usuario:

- Actor: Usuario (Estudiante o Profesor)
- Descripción: El usuario puede ver sus propios datos, como nombre, apellido y dirección de correo electrónico.

Gestión de Estudiantes:

- Actor: Gestor de Estudiantes
- Descripción: El gestor de estudiantes puede agregar, modificar o mostrar información de estudiantes, como nombre, apellido, código y notas.

Gestión de Profesores:

- Actor: Gestor de Profesores
- Descripción: El gestor de profesores puede agregar, modificar o mostrar información de profesores, como nombre, apellido, código, especialidad, etc.

Registro de Notas:

- Actor: Profesor
- Descripción: El profesor puede ingresar y actualizar las notas de los estudiantes por asignatura.

Guardar y Cargar Datos:

- Actor: Base de Datos
- Descripción: El sistema debe ser capaz de guardar y cargar datos de estudiantes y profesores de la base de datos.

Consulta de Notas:

- Actor: Estudiante
- Descripción: El estudiante puede ver sus propias notas.

Mantenimiento del Sistema:

- Actor: Administrador
- Descripción: El administrador puede realizar tareas de mantenimiento, como actualizar el sistema, corregir errores y realizar mejoras.