<!--Proyecto 01 Python-->

Registro de Asistencia {

<Por="Anderson Stiven Galvis"/>

}

Contenidos

Introducción {

ACME Education, una institución comprometida con la formación técnica y profesional, ha decidido modernizar su sistema de gestión de asistencia para mejorar la eficiencia y precisión en el seguimiento de la asistencia de sus estudiantes. El Sistema de Gestión de Asistencia Académica (SISGESA) se desarrollará en Python como una aplicación de consola, permitiendo a los usuarios interactuar de manera sencilla con diversas funcionalidades, desde el registro de asistencia hasta la generación de informes. Este proyecto busca no solo automatizar el proceso de registro, sino también proporcionar herramientas que faciliten la toma de decisiones basadas en datos confiables.

- Estructuración.
- Revisión de errores.
- Corrección.
- Formateo.
- Limpieza.
- Exportación.
- Depuración.

}

Objetivos{

Automatizar el Registro de Asistencia:

Desarrollar un sistema que permita registrar de manera automatizada la asistencia de los estudiantes mediante un menú interactivo en consola,

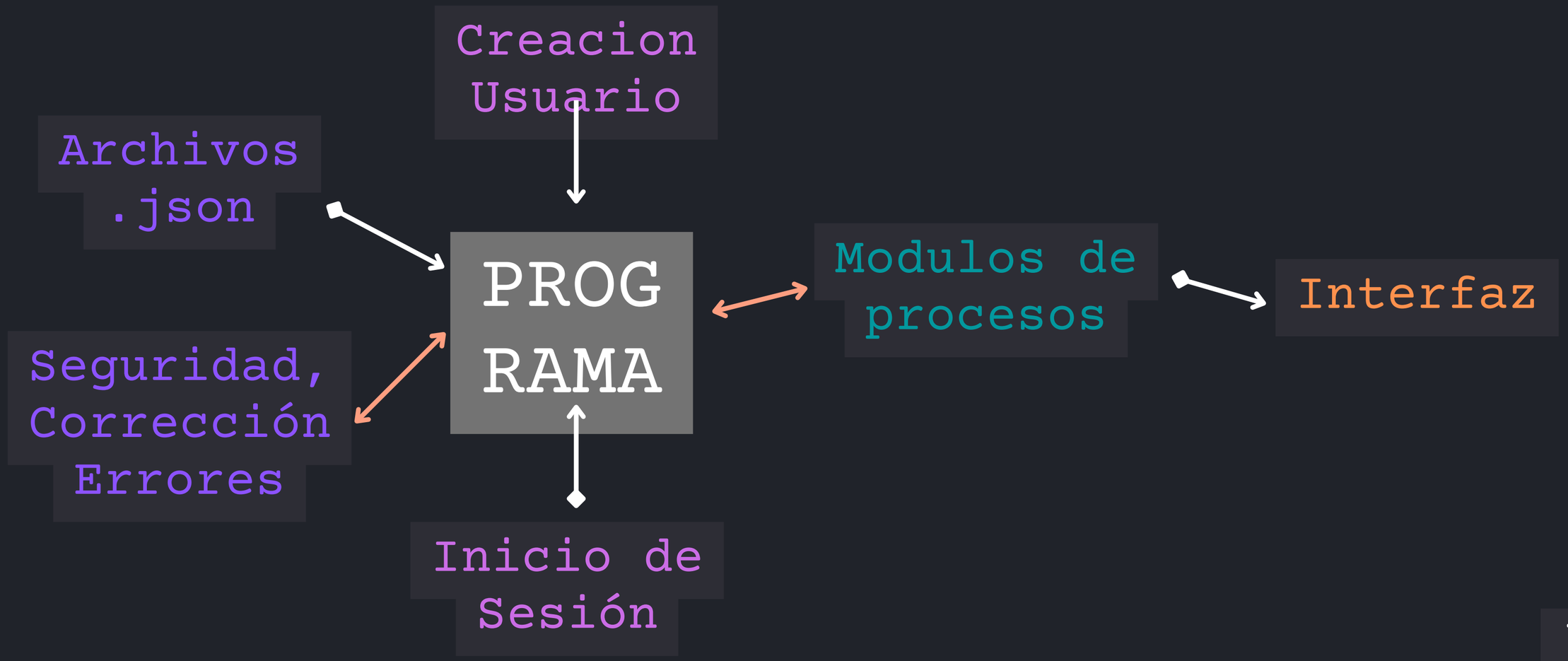
Generación de Informes Efectivos:

Implementar funcionalidades que permitan generar informes detallados sobre la asistencia.Estos informes se presentarán en un formato legible y estructurado, facilitando su análisis.

Implementar una funcionalidad que permita registrar y gestionar la asignación de estudiantes a grupos y módulos de manera sencilla y eficiente. Además, el sistema ofrecerá opciones de consulta que permitirán a los usuarios acceder rápidamente a información relevante.

}

Estructura {


Creacion Usuario
Archivos .json
PROG RAMA
Modulos de procesos
Interfaz
Seguridad, Corrección Errores
Inicio de Sesión

}

Puntos clave {

01

Automatización Registro asistencia.

02

Interfaz de usuario Amigable.

03

Corrección de errores.

04

Optimizar el rendimiento general.

05

Generación de informes detallados.

06

Gestión efectiva de datos.

07

persistencia de datos.

08

Limpiar la base de datos.

}

Aspectos de mejora {

Persistencia de datos.

- **Continuidad del Sistema: La persistencia de datos asegura que toda la información registrada, como asistencia, grupos, módulos y usuarios, se mantenga disponible incluso después de cerrar el programa. Esto es esencial para un funcionamiento fluido y continuo del sistema.**
- **Facilitación del Análisis: Al almacenar datos de manera estructurada, se facilita la generación de informes y consultas, lo que permite a los administradores analizar patrones y tendencias en la asistencia.**

Seguridad.

- **Encriptación de Contraseñas: Todas las contraseñas se almacenan de forma segura utilizando el algoritmo SHA-256, asegurando que la información sensible esté protegida contra accesos no autorizados.**
- **Acceso Autorizado: El sistema requiere que los usuarios ingresen una contraseña al iniciar, garantizando que solo las personas autorizadas puedan acceder a la información.**

}

Conclusión {

La implementación de bibliotecas nativas y el uso de archivos JSON para la persistencia de datos garantizan que el código se mantenga ligero y accesible, evitando la necesidad de instalaciones adicionales. Esto no solo mejora la seguridad y la estabilidad del sistema, sino que también facilita su escalabilidad y adaptación a futuras necesidades. Además, el manejo de errores se ha integrado cuidadosamente, asegurando que el sistema pueda recuperarse de situaciones inesperadas sin interrumpir la experiencia del usuario.

```python
import hashlib #importamos el diccionario
from modulo.persis import guardarDat, cargarDat, CargarUser, guardarUser
from modulo.interfazz import menu, Entrada

#fucnion para encriptar la contraseña
def passwDencrip(contra):…

#Creacion del usuario para su primer ingreso
def CrearUser(User):…

#inica la funcion en el sofwar



def inicSesion(User):…



def gestionUser (User):…


def agregarEstud(datos):…


def agregarProfe(datos):…

def agregarMod(datos):…
#se piden los datos para los grupos
def agregarGrupo(datos):…

def VerifiCod(cod, datos):…


def cambiarPsw(User):…



def menuPrin(datos):…

```

}

<!--Proyecto 01 Python-->

Gracias {

<Por="Anderson galvis"/>

}