Documentación

Conocimiento en Python y sus librerías

Librerías:

* Time: Un módulo hecho a partir de Python por Guido Van Rossum, sirve para analizar horarios y el tiempo ya que cuenta cada segundo que paso desde su programación para determinar la fecha y hora actual. Gracias a esto podemos utilizarlo en nuestro programa de asistencias para calcular llegadas tarde, calcular el valor correspondiente de las faltas, retiros, etc.
* Mysql.connector.python: Una librería que cumple la función de puente entre el programa de Python y la base de datos preparada con MySQL en PHPMyAdmin mediante el usuario y la dirección IP que comparten

Puntos importantes de Python considerados para su programación:

Programación Orientada a Objetos (POO)

La programación orientada a objetos esta fundamentada por cuatro pilares principales:

* Abstracción: Asignar métodos y atributos a clases los cuales al ser rellenados con información dan lugar a distintos objetos. Esto nos permite optimizar el código y crear varios elementos distintos a base del mismo bloque de código
* Polimorfismo: Por mas que varios objetos sean codificados de la misma manera, cada uno responderá dependiendo de la información que tiene o que conoce por lo que podemos dar instrucción sin preocuparnos porque la ejecución de varios elementos termine en un error
* Herencia: Las clases creadas pueden transmitir funciones a otras clases “hijas” para optimizar el código y hacerlo mas eficiente
* Encapsulación: Los objetos no podrán acceder a datos sobre los que no tengan relación o no les correspondan por lo que aumenta la seguridad y se elimina el riesgo de que algo o alguien modifique información que no le es licito

**Cambio de lenguajes**

Dejamos Python debido a que ninguna de las herramientas que ofrece nos resultaron útiles para comunicarnos con el dispositivo de lectura de huellas “DigitalPersona”. En su lugar nos informamos sobre los lenguajes de programación que permiten el funcionamiento de dicho dispositivo y elegimos uno de ellos: C#.

C# nos resulto conveniente ya que cumple con los requisitos que establecimos para crear la aplicación:

* + - Capacidad de conexión con una base de datos hecho en MySQL
    - Interfaz grafica sencilla de comprender y de programar
    - Compatibilidad con el lector de huella

**Pasos para trabajar:**

1. Backend: Necesita esperar que se termine la base de datos en MySQL para programar el funcionamiento correctamente
2. Aprender como identificar una huella digital, guardarla en la BD y recopilarla más tarde
3. Ponerle contraseña a la base de datos
4. Pensar Funciones para manipular la tabla del salón, años y divisiones: a los alumnos y profesores
   1. Conseguir hora de ingreso y egreso
   2. Conseguir los horarios de un alumno o grupo especifico
   3. Conseguir la división y/o grupo al que pertenece un alumno
   4. Identificar turnos a los que pertenece un alumno
   5. Mostrar al alumno salón, materia y profesor que tendrá en la próxima hora
   6. Saber si un alumno quedo libre
   7. Saber que preceptor está a cargo de un grupo o alumno especifico

**Guía para completar el segundo paso de trabajo:**

1. Conectar la base de datos de MySQL (Para esto vamos a usar una de prueba). La estructura de la base de datos va a ser una sola tabla con 6 atributos: ID, cuatro huellas digitales
2. Francia
3. Reemplazar las variables de DPFP.Templates que se usan en el código con las huellas de la BD
   1. El DPFP.templates que necesitamos reemplazar es la variable de iteración del foreach que esta en la línea 21 del archivo VerificationForm.cs (La línea de codigo que se menciona no considera futuras modificaciones).

**Preguntas para entrevista:**

1. Cuándo un alumno se retira, ¿se le aplica una cantidad de falta fija o se aplica dependiendo del contexto del retiro?
2. ¿Qué datos se necesitan para confirmar un retiro? ¿Cómo corroboramos que la persona que hace el retiro del alumno esta avalada o registrada con permiso para hacer el retiro del alumno? ¿Dónde queda registrado?