

Programación Orientada a Objetos – Unidad 5

Caso de estudio: Aplicación web de Control de Asistencias en un Colegio

Descripción

El colegio secundario “San Miguel” plantea como estrategia de control en los estudiantes manejar el registro de asistencia de cada estudiante a través de una aplicación web de modo que los padres puedan acceder en cualquier momento a consultar las inasistencias de sus hijos.

La aplicación tendrá como usuarios a los preceptores, como así también a los padres de los estudiantes.

El colegio tiene cursos desde primero a sexto año, de cada uno hay dos divisiones (12 cursos en total).

El colegio tiene como reglamentación que la inasistencia a clase de aula se computa como una falta y la inasistencia a clase de educación física se computa como media falta.

El objetivo es que cada día, el preceptor ingrese a la aplicación con su correo y clave, y registre la asistencia de los estudiantes. Además, el preceptor debe poder obtener un informe detallado que, para cada estudiante que asiste a un curso dado, indique: cantidad de asistencia (si asistió) a clases de aula y a clases de educación física, cantidad de inasistencias a clases de aula justificadas e injustificadas y cantidad de inasistencias a clases de educación física justificadas e injustificada y cómputo total de inasistencias. Otro informe requerido es un listado de asistencia a una clase (aula o educación física) de un curso para una fecha ingresada, donde por cada estudiante se indique si asistió o no.

De una asistencia de un estudiante se registra la fecha, el tipo de clase (1- aula o 2- educación física), la asistencia del estudiante (‘s’- si está presente o ‘n’ – no está presente) y justificación (en caso de inasistencia de que la inasistencia este justificada por salud, viaje, competencia, etc).

Por cada estudiante se registra solo uno de sus padres (padre, madre o tutor). El padre, madre o tutor registrado ingresará a la aplicación con su correo y clave, y solo podrá consultar la información de un estudiante que sea su hijo. Debe poder ingresar el DNI del estudiante y obtener un informe de inasistencias donde conste: fecha, clase (aula o educación física) y justificación (si la inasistencia está justificada) de cada inasistencia, y al final el cómputo total de inasistencias.

Historias de usuario

FUNCIONALIDAD #1: Acceso a la aplicación

Descripción

COMO	Usuario Registrado
NECESITO	Acceder a la aplicación
PARA	Hacer uso de las funcionalidades que la misma provee

Escenario

#1		Comentarios
	El usuario ingresa su correo, contraseña y rol (preceptor o padre) con el cual va a usar la aplicación.	Los caracteres de la contraseña no se deben ver al momento del ingreso.

		Los usuarios deben estar previamente registrados.
	La aplicación verifica los datos ingresados.	
	Si el correo y la contraseña introducidos corresponden a un usuario registrado con el rol solicitado, la aplicación da acceso a las funcionalidades que correspondan.	
	Si el correo y la contraseña introducidos no corresponden a un usuario registrado con el rol solicitado, la aplicación muestra un mensaje de error.	

FUNCIONALIDAD #2: Registrar asistencia de un curso

Descripción

COMO	Preceptor registrado
NECESITO	Acceder a la aplicación
PARA	Registrar la asistencia de un curso

Escenario

#2		Comentarios
	El preceptor selecciona la opción "Registrar asistencia".	
	El preceptor selecciona el curso, la clase (aula o educación física) y la fecha para cargar la asistencia.	Se listan solo los cursos asignados al preceptor.
	La aplicación muestra los estudiantes del curso seleccionado.	Los estudiantes deben mostrarse ordenados alfabéticamente.
	El preceptor ingresa la asistencia para cada estudiante y luego pulsa "Guardar asistencia"	Se usará el valor "n" para indicar que no asistió y el valor "s" para indicar que sí asistió. A la clase de aula le corresponde el código 1 y a la clase de educación física el código 2. La justificación queda en vacío en caso de que la inasistencia no esté justificada.
	La aplicación registra la asistencia de cada estudiante.	

FUNCIONALIDAD #3: Obtener informe detallado

Descripción

COMO	Preceptor Registrado
NECESITO	Acceder a la aplicación
PARA	Obtener un informe detallado de asistencias e inasistencias de los estudiantes de un curso

Escenario

#3		Comentarios
	El usuario selecciona la opción "Informe con detalles".	
	El preceptor selecciona el curso del cual requiere el informe.	Se listan solo los cursos asignados al preceptor.
	La aplicación muestra, de cada estudiante cantidad de clases de aula presente, cantidad de clases de educación física presente, cantidad de clases de aula ausente justificadas, cantidad de clases a clases de aula ausente injustificadas, cantidad de clases de educación física ausente justificadas, cantidad de clases a clases de educación física injustificadas y cómputo de la cantidad total de inasistencias.	El listado debe emitirse ordenado alfabéticamente. Para el cómputo de la cantidad total de inasistencias se debe considerar que la inasistencia a clase de aula es 1 falta y la inasistencia a clase de educación física es media falta.

FUNCIONALIDAD #4: Obtener listado de asistencia a clase

Descripción

COMO	Preceptor Registrado
NECESITO	Acceder a la aplicación
PARA	Obtener un listado de asistencia a un tipo de clase para un curso en una fecha dada.

Escenario

#4		Comentarios
	El usuario selecciona la opción "Listado de asistencia".	
	El preceptor selecciona el curso, la clase y la fecha del cual requiere el informe.	La clase puede ser aula o educación física
	La aplicación muestra un listado con el nombre de cada estudiante y si estuvo presente o ausente en la clase y fecha dada.	El listado debe emitirse ordenado alfabéticamente.

FUNCIONALIDAD #5: Consultar inasistencias de estudiante

Descripción

COMO	Padre Registrado
NECESITO	Acceder a la aplicación
PARA	Consultar las inasistencias de un hijo.

Escenario

#5		Comentarios
	El padre selecciona la opción "Consultar inasistencias".	
	La aplicación solicita que ingrese el DNI del estudiante.	
	La aplicación muestra fecha, tipo de clase y justificación (si la inasistencia está justificada) de cada inasistencia, y al final el cómputo total de inasistencias.	

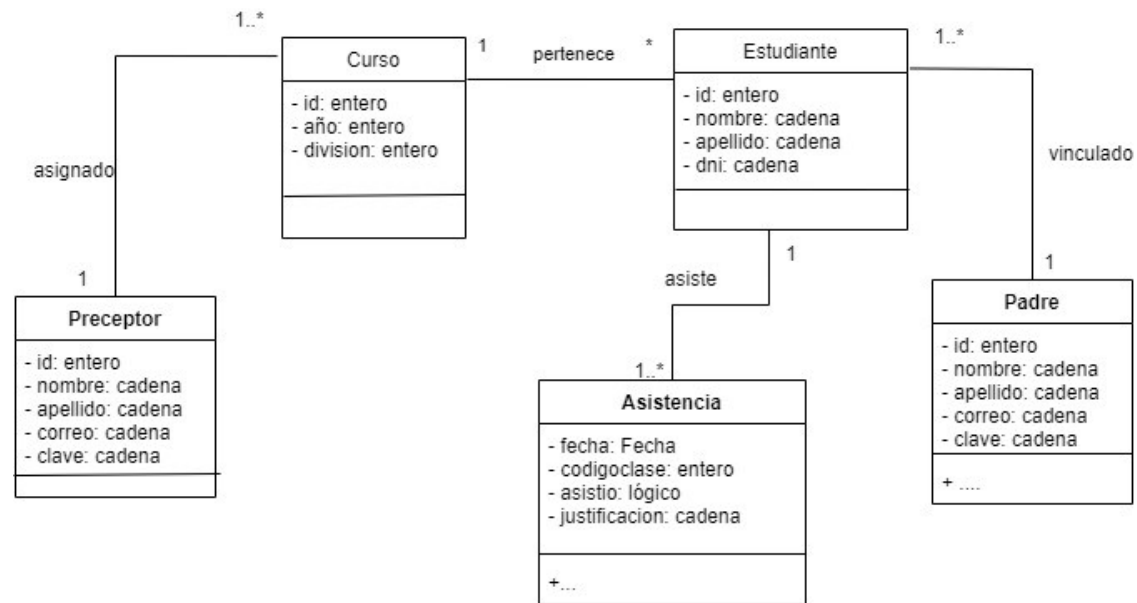
Usuarios

Los usuarios de la aplicación son los preceptores y padres registrados.

Reglas de negocio globales

1. Un preceptor tiene asignado uno o más cursos.
2. Un curso tiene asignado solo un preceptor.
3. A un curso pertenecen muchos estudiantes.
4. Un estudiante pertenece solo a un curso.
5. Un padre es vinculado a uno o más estudiantes.
6. Un estudiante es vinculado a un padre.
7. Los tipos de clases están codificados, correspondiendo a la clase de aula el código 1 y a la clase de educación física el código 2.
8. Una inasistencia a clase de aula se computa como una falta.
9. Una inasistencia a clase de educación física se computa como media falta.
10. En todas las páginas de la aplicación debe figurar el nombre del colegio.

Diagrama de clases



Tablas de base de datos

- Preceptor

id	nombre	apellido	correo	clave
----	--------	----------	--------	-------

Nota: la clave de los usuarios que ya están guardados en la base de datos, es el nombre de usuario del correo electrónico (anaestrada, luchicortez, martosj, gimeaballay67).

- Curso

id	anio	division	idpreceptor
----	------	----------	-------------

- Estudiante

id	nombre	apellido	dni	idcurso	idpadre
----	--------	----------	-----	---------	---------

- Asistencia

id	fecha	codigoclase	asistio	justificacion	idestudiante
----	-------	-------------	---------	---------------	--------------

- Padre

id	nombre	apellido	correo	clave
----	--------	----------	--------	-------

Nota: la clave de los usuarios que ya están guardados en la base de datos, es el nombre de usuario del correo electrónico (rodalb, carmor, luicar, etc)

Consideraciones para la resolución

1. Definir las clases correspondientes al diseño planteado.
2. Diseñar páginas fáciles de usar, comprensibles y agradables para el usuario.
3. Para ingresar y mostrar datos utilizar los controles apropiados.
4. Usar Flask y SQLAlchemy para manejar desde la aplicación los datos almacenados en la base de datos.
5. Verificar la validez de los datos (fechas, números, etc.) ingresados y de los datos que se almacenan en la base de datos.
6. Las contraseñas de los usuarios se deben almacenar cifradas por razones de seguridad.

Para cifrar las contraseñas pueden utilizar el siguiente código:

```
import hashlib
if __name__ == "__main__":
    # cifrado de la clave utilizando md5
    clave = 'julia459'
    result = hashlib.md5(bytes(clave, encoding='utf-8'))
    # muestra la clave cifrada en hexadecimal, esta es la que se guarda en base de datos
    print(result.hexdigest())
```

Nota: las contraseñas **no se descifran**. Se ingresa una contraseña, se cifra, el valor obtenido se compara con el valor registrado en la base de datos.