TP5 - Consigna

Pensamiento en clases y objetos

Como para ir introduciéndonos en los conceptos de clases y objetos a través del análisis de diferentes contextos, pero también para ir viendo como otras personas ven la misma cosa.

Importante: Este práctico tiene una mecánica diferente al resto y se desarrollará en tres fases.

- 1. La primera finaliza luego de la clase del lunes (8/4 21:00hs), en esta ustedes deberán completar el análisis de alguna situación que hayan elegido.
- 2. Luego del lunes, hasta el jueves a la noche (11/4 22:00hs), deberán revisar algunas de las entregas de sus colegas.
- 3. Al final, analizaremos en clase algunos interesantes, pero los publicaré en el repositorio de la cursada para que queden visibles para todos.

El resultado del TP se obtendrá basándose en: su entrega, las revisiones que reciban y los ajustes al resultado que sean necesarios. Por más que vean números, el resultado será del mismo tipo que los otros TP.

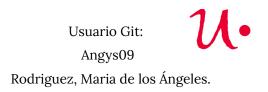
Descripción:

- 1. **Elección de contexto:** Elijan uno de los contextos de la lista al final de este enunciado.
- 2. **Describe el contexto**: Indique como y donde le darían uso (recuerden los ejemplos con respecto al auto que vimos el miércoles)
- 3. **Identificación de partes:** Describe las partes principales que componen el elemento en el contexto elegido.
- 4. **Información:** Define qué información se guarda en cada una de las partes identificadas.
- 5. **Comportamiento:** Describe las acciones que se pueden realizar con el elemento en el contexto elegido. Si es posible, indiquen cómo cambiaría la información contenida.

Recomendaciones:

- No pensar en términos de programación, sino en la descripción del elemento y sus funcionalidades en el contexto elegido.
- Enfocarse en las características esenciales del elemento y no en detalles superfluos.
- Ser creativo y pensar en diferentes operaciones que se puedan realizar con el elemento.

Universidad de Río Negro **Ingeniería en Computación** Programación II 2024 - TP 5



El contexto elegido fue el de la gestión de una cita médica en un consultorio, clínica u hospital.

Las Citas médicas, o 'Turnos' forman parte de un proceso fundamental para los centros de salud, permiten a los pacientes programar consultas con un médico de la especialidad que elijan, en en determinado momentos para ambas partes, esto mejora la eficiencia y la satisfacción tanto del paciente como del personal médico.

Las partes que componen el elemento de este contexto son:

- Fecha y hora de la cita.
- Datos del paciente que solicita la misma.
- Datos del médico que atenderá la cita.
- Especialidad médica del médico.

En cada parte identificada se guardará la siguiente información:

- Fecha y hora de la cita: momento en que se llevará a cabo la consulta médica.
- Datos del paciente: Incluye el nombre y otros datos de contacto del paciente.
- Datos del médico: Incluye nombre y otros datos de contacto del médico.
- Especialidad médica: indica la especialización del médico tratante.

Las acciones que pueden hacerse en este contexto son:

- Programar una cita médica: Permite a un paciente solicitar una cita con un médico en una fecha y una hora determinada.
- Cancelar la cita médica: Permite al paciente cancelar una cita programada previamente.

Universidad de Río Negro **Ingeniería en Computación**

Usuario Git: Angys09



Programación II 2024 - TP 5

Rodriguez, Maria de los Ángeles.

- Modificar una cita médica: Permite realizar cambios en la fecha, hora o médico asignado para una cita.
- Consultar citas programadas: Permite al personal médico o administrativo ver las citas programadas para un médico en particular o una fecha específica.

CitaMedica

- fecha: LocalDate

- hora: int

- paciente: Persona

- medico: Persona

- especialidad: String

+ CitaMedica()

+ CitaMedica(fecha: LocalDate, hora: int, paciente:

Persona, medico: Persona, especialidad: String)

+ programarCita(): String

+ modificarCita(): String

+ cancelarCita(): boolean

+ ConsultarCita() : String

Nota: después de analizar la teoría de POO, me di cuenta de que podemos encapsular los datos de paciente y médico, en otra clase llamada Persona, esta tendrá sus propios métodos y atributos, que sirven para sumarlos a esta clase.