

|  |  |  |
| --- | --- | --- |
| **Nombre**:  Abraham Basurto Alonso | | **Matrícula**:  AL02885896 |
| **Nombre del curso:**  Computacion en Java | **Nombre de la profesora**:  Ing.Claudia Lorenzo Muradas | |
| **Módulo:**  Modulo 01 | **Actividad: evidencia parte 1 avance** | |
| **Fecha**: 10/02/2023 | | |
| **Bibliografía**:  <https://app.creately.com/d/IMR4htIBeQx/edit> <https://cursos.tecmilenio.mx/courses/151490/pages/mi-curso?module_item_id=542121> <https://github.com/AbBasurto?tab=repositories> | | |

**2. Diagrama de flujo**

Elaborar un diagrama de flujo del programa que cubra los requerimientos previamente mencionados.  
Diagrama

Descripción generada automáticamente

**3. Diseño del programa (diagrama de clases)**

Después de avanzar en los conocimientos sobre programación orientada a objetos, se realizará un diagrama de clases

donde se desglosarán los componentes de la aplicación. Se deberán estructurar los componentes de acuerdo con las funcionalidades del sistema, por ejemplo:

• Clase Principal.

• Clase para Doctor.

• Clase para Paciente.

• Clase para Cita.

El reto es pensar en otras clases, tanto abstractas como concretas, así como posibles interfaces que sean necesarias para

implementar correctamente las funcionalidades del programa, no olvidar tomar en cuenta los conceptos de herencia y

polimorfismo.

**4. Pseudocódigo**

Con base en el diagrama, traducirlo a pseudocódigo. Si es necesario, mejorar el diagrama de flujo:  
• **Dar de alta doctores y pacientes.**  
-Crear menú principal para poder registrar usuario según sea necesario Doctores/Administradores/Pacientes, para ambos generar: usuario (nombre completo, sexo y edad), contraseña (6 dígitos) y estatus (doctor/administrador/paciente).   
-Si es doctor o administrador otorgar permisos de sistema para ver y editar citas.

• **Crear una cita con fecha y hora.**  
-Si es paciente otorgar permisos para generar cita con fecha y hora (sujeto a disponibilidad).  
-el paciente ingresará al sistema sus datos y contraseña con lo cual podrá generar y ver las citas disponibles.  
-Si la contraseña es equivocada regresar al inicio y volver a pedir las claves.

**• Relacionar una cita con un doctor y un paciente.**  
-Implementar método para relacionar cita entre “usuario paciente y usuario doctor”

• **Tener control de acceso mediante administradores, solo ciertos usuarios podrán acceder al sistema mediante un identificador y una contraseña.**  
-Usuarios doctor y administrador tienen privilegios para modificar y revisar sistema  
-Siempre y cuando ingresen con el usuario correcto, si la contraseña es equivocada regresar al inicio y volver a pedir las claves.