Universidad Rafael Landívar

Facultad de Ingeniería

Ingeniería en Informática y Sistemas

Programación Web

Definición de Proyecto

Nombre de proyecto: SelfMedX

El nombre SelfMedX hace referencia a la no automedicación, el proyecto propone enlazar a un médico con un usuario para otorgarle una guía de consumo de medicamentos que no requieren de prescripción médica.

En primera instancia el sistema funcionará para gestionar y controlar citas entre médicos y pacientes. Mientras el proyecto crezca se buscará un modelo de negocio B2C

El sistema tendrá:

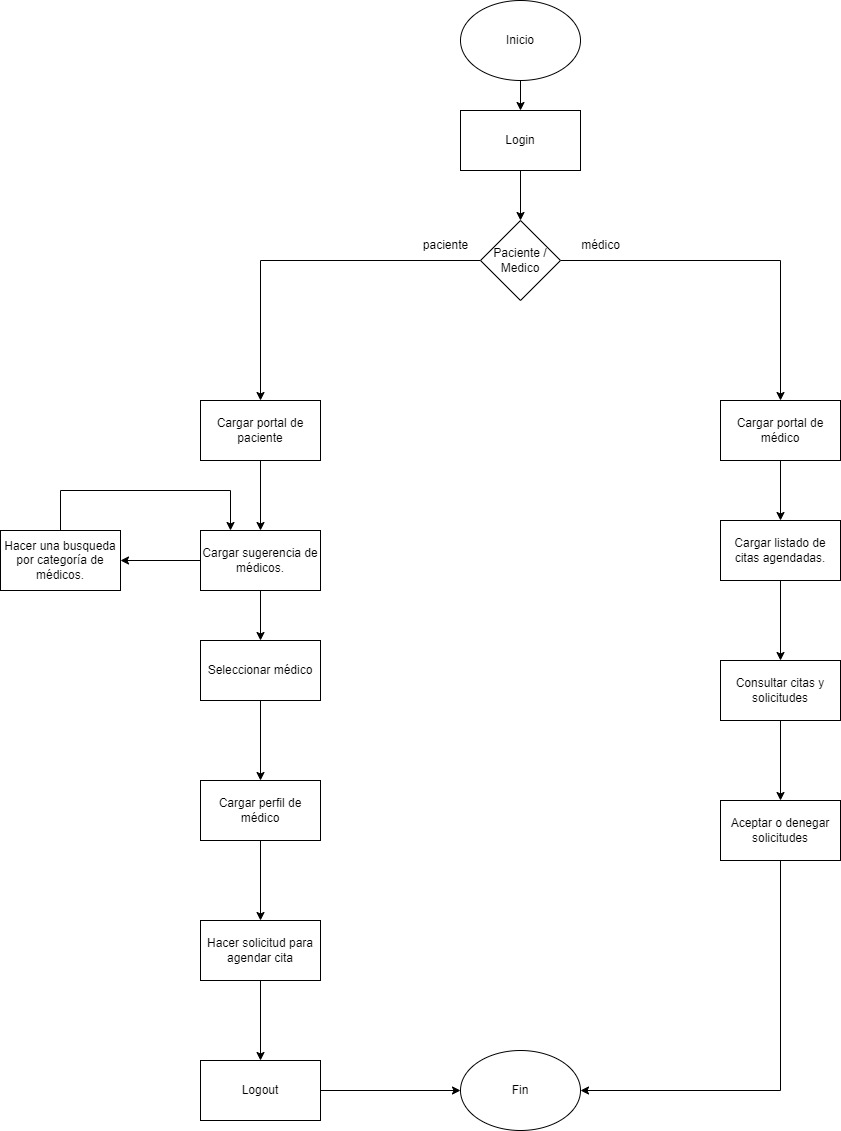
* Poseer Login
  + Para pacientes (seguimiento y solicitud de citas) y para médicos (Sistema de control de citas)
* Mantenimiento de Altas/Modificaciones/Bajas de usuarios
  + De información de citas y de los usuarios.
  + Si un usuario ya no quiere seguir usando la aplicación podrá eliminar su cuenta.
* Identificación de usuarios y roles
  + Habrá dos roles (en primera instancia), el rol de usuario paciente y el rol “médico” que estará asociado al sistema de citas de un “consultorio”.
* Interactuar con un API Rest para conectarse a la base de datos utilizando entity framework y brindar seguridad de comunicación
  + Se planea usar una base de datos relacional a la cual el api se conectará.
* El contenido a mostrar debe responsivo
* Administrar y gestionar diferentes eventos (errores, advertencias y avisos)

Stack de desarrollo:

Base de datos: MySql o SqlServer

Backend: dotNet Core

FrontEnd: dotNet core con Bootstrap



Descripción de entidades

Tabla usuario

id (int): Identificador único del usuario.

Nombres (string): Nombre del usuario.

Apellidos (string): Apellidos del usuario.

FechaNacimiento (DateTime): Fecha de nacimiento del usuario.

FechaCreacion (DateTime): Fecha en que se creó el registro del usuario.

FechaElimina (DateTime, nullable): Fecha en que se eliminó el registro del usuario (si es que se eliminó).

Vigente (bool, nullable): Indica si el registro del usuario está vigente.

UrlImg (string, nullable): URL de la imagen del usuario.

Tabla paciente

id (int): Identificador único del paciente.

id\_usuario\_paciente (int): Identificador del usuario asociado al paciente.

Tabla medico

id (int): Identificador único del médico.

id\_usuario\_medico (int): Identificador del usuario asociado al médico.

tituloCorto (string): Título corto del médico.

Tabla cita

id (int): Identificador único de la cita.

FechaCreacion (DateTime): Fecha en que se creó el registro de la cita.

FechaCita (DateTime): Fecha y hora de la cita.

idMedico (int): Identificador del médico asociado a la cita.

idPaciente (int): Identificador del paciente asociado a la cita.

Estado (int): Estado de la cita.

Descripcion (string, nullable): Descripción de la cita.

Tabla entidadmedica

id (int): Identificador único de la entidad médica.

nombre (string): Nombre de la entidad médica.

direccion (string, nullable): Dirección de la entidad médica.

Tabla entidadmedica\_medico

id\_entidad (int): Identificador de la entidad médica.

id\_medico (int): Identificador del médico.

Tabla historialmedico

id (int): Identificador único del historial médico.

idPaciente (int): Identificador del paciente asociado al historial médico.

enfermedad (string): Nombre de la enfermedad.

tratamiento (string, nullable): Tratamiento de la enfermedad.

fechaingreso (string): Fecha de ingreso del historial médico.

Casos de uso esperados:

Caso de uso 1: Sistema de control de citas - Médico hace login al sistema a través de la pantalla de login

Actores principales: Médico.

Actores secundarios: Sistema.

Precondiciones:

El médico debe tener una cuenta en el sistema.

La pantalla de login debe estar disponible.

Flujo principal:

El médico accede a la pantalla de login del sistema.

El médico ingresa su nombre de usuario y contraseña en los campos correspondientes.

El sistema verifica la validez de las credenciales ingresadas.

El sistema muestra la página principal del sistema al médico.

Flujo alternativo:

Si el médico ingresa credenciales inválidas, el sistema muestra un mensaje de error y solicita que se ingresen nuevamente las credenciales.

Postcondiciones:

El médico puede acceder a las funcionalidades del sistema a las que tiene permiso según su rol.

Diagrama

Descripción generada automáticamente

Caso de uso 2: El médico puede ingresar una cita de paciente según el script compartido con anterioridad en este chat.

Nombre: Ingresar cita de paciente

Actores: Médico

Descripción: Este caso de uso describe cómo el médico puede ingresar una nueva cita para un paciente en el sistema. El médico debe tener una sesión iniciada en el sistema para poder acceder a la funcionalidad de ingreso de citas.

Flujo básico:

El médico accede a la funcionalidad de ingreso de citas.

El sistema muestra un formulario para ingresar los datos de la cita: fecha, hora, paciente, médico, motivo de la cita, entre otros.

El médico ingresa los datos correspondientes en el formulario.

El sistema valida los datos ingresados y muestra un mensaje de confirmación al médico.

El médico confirma la creación de la cita.

El sistema almacena la cita en la base de datos.

Flujos alternativos:

Si el médico ingresa datos inválidos en el formulario, el sistema muestra un mensaje de error y solicita que se corrijan los datos.

Si el médico cancela la creación de la cita, el sistema no la almacena en la base de datos.

Si el médico intenta ingresar una cita para un paciente que ya tiene una cita programada en el mismo horario, el sistema muestra un mensaje de error y solicita que se seleccione una hora diferente para la cita.

Precondiciones:

El médico debe tener una sesión iniciada en el sistema.

El paciente debe estar registrado en el sistema.

Postcondiciones:

La cita queda registrada en el sistema y asociada al paciente correspondiente.

Diagrama

Descripción generada automáticamente

Caso de uso 3: Se puede cambiar la fecha de una cita.

Nombre: Cambiar fecha de cita

Actores: Recepcionista, Médico

Descripción: El actor Recepcionista o Médico pueden cambiar la fecha de una cita programada.

Flujo principal:

El actor Recepcionista o Médico inicia sesión en el sistema.

El actor Recepcionista o Médico busca la cita a la que desea cambiar la fecha.

El actor selecciona la opción de cambiar la fecha de la cita.

El actor ingresa la nueva fecha y hora para la cita.

El sistema verifica la disponibilidad de la nueva fecha y hora en el calendario del médico.

Si la fecha y hora están disponibles, el sistema guarda la nueva fecha y hora para la cita.

El sistema envía una notificación al paciente informando el cambio de fecha de la cita.

Flujos alternativos:

A1. Si la fecha y hora seleccionadas no están disponibles, el sistema muestra un mensaje de error y solicita al usuario que seleccione otra fecha y hora disponibles.

A2. Si el actor no puede encontrar la cita que desea modificar, puede buscarla utilizando la función de búsqueda del sistema.

A3. Si el actor desea cancelar la modificación de la cita, puede seleccionar la opción de cancelar y el sistema vuelve a la pantalla de citas programadas.

Precondiciones:

El actor debe estar registrado en el sistema y haber iniciado sesión.

La cita debe existir en el sistema.

Postcondiciones:

El sistema actualiza la fecha y hora de la cita en la base de datos.

El sistema envía una notificación al paciente informando el cambio de fecha de la cita.

Diagrama

Descripción generada automáticamente