

Integrantes: Juan Gallardo, Amable Quishpe, Kevin Chicaiza

Fecha: 26/1/2025

Introducción

Un barrio de Quito ha solicitado sistematizar el proceso de facturación y gestión del consumo de agua donde cada familia tiene un medidor individual, ya que el sistema manual o no centralizado ha generado demasiadas deficiencias.

El programa H2O UIO tiene como objetivo principal identificar y gestionar el pago del consumo de agua de cada familia a través de una aplicación web. Cada familia se abastece de agua a través de medidores individuales, y el programa se encarga de registrar la medida mensual de cada medidor.

La administración del sistema estará a cargo de la Empresa de Agua Potable, quienes podrán realizar cobros, crear nuevos usuarios y añadir las medidas mensuales. Otros administrativos tendrán acceso solo para visualizar la base de datos, sin permisos para modificarla o realizar cobros.

De esta manera, existirá mayor fluidez y facilidad para realizar los cobros de parte de la Empresa Pública y los pagos de parte de los usuarios, ya que no será necesaria la impresión de papeletas físicas de agua para todos los usuarios debido a que toda la información requerida en tiempo real se encontrará en la página web.

Descripción del Proyecto

El proyecto es una aplicación desarrollada en Visual Studio 2022 con un enfoque integral que abarca un backend robusto y un frontend amigable. Este sistema está diseñado para la gestión eficiente del consumo de agua potable y el cobro de tarifas, atendiendo las necesidades de administradores y clientes.

Nuevas Implementaciones:

1. Patrón MVVM:

- En la aplicación móvil desarrollada con .NET MAUI, se ha implementado el patrón Model-View-ViewModel (MVVM) para separar la lógica de negocio de la interfaz de usuario. Esto garantiza una arquitectura escalable y fácil de mantener.

2. API Local:

- Una API local basada en ASP.NET Core publica los datos del backend (SQL Server) para que sean consumidos por la aplicación móvil.
- Incluye endpoints RESTful que soportan operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para usuarios, medidas y cobros.

3. API Pública:

- Se integra una API pública relacionada con estadísticas de consumo y tarifas de agua potable.
- La aplicación utiliza esta API para mostrar información relevante, como comparativas de consumo promedio y tarifas por región.

4. Base de Datos Local (SQLite):

- La aplicación móvil incluye una base de datos local (SQLite) que permite operaciones CRUD.
- Esta base de datos almacena temporalmente datos obtenidos de la API pública, asegurando el funcionamiento offline de la aplicación.

Tabla de Solución

La tabla de solución está constituida por dos componentes principales:

Aplicación Web MVC:

- **Modelo (Model):** La aplicación utiliza el enfoque Code First, donde las clases de C# representan las entidades y relaciones de la base de datos. Esto permite crear y actualizar automáticamente la base de datos utilizando migraciones.
- **Vista (View):** Presenta interfaces dinámicas para que los administradores interactúen con los datos de manera clara y estructurada.
- **Controlador (Controller):** Maneja las solicitudes del usuario, procesa la lógica del negocio y devuelve las vistas o respuestas adecuadas.
- **Propósito:** Actúa como el sistema principal para gestionar usuarios, medidas de consumo y cobros.

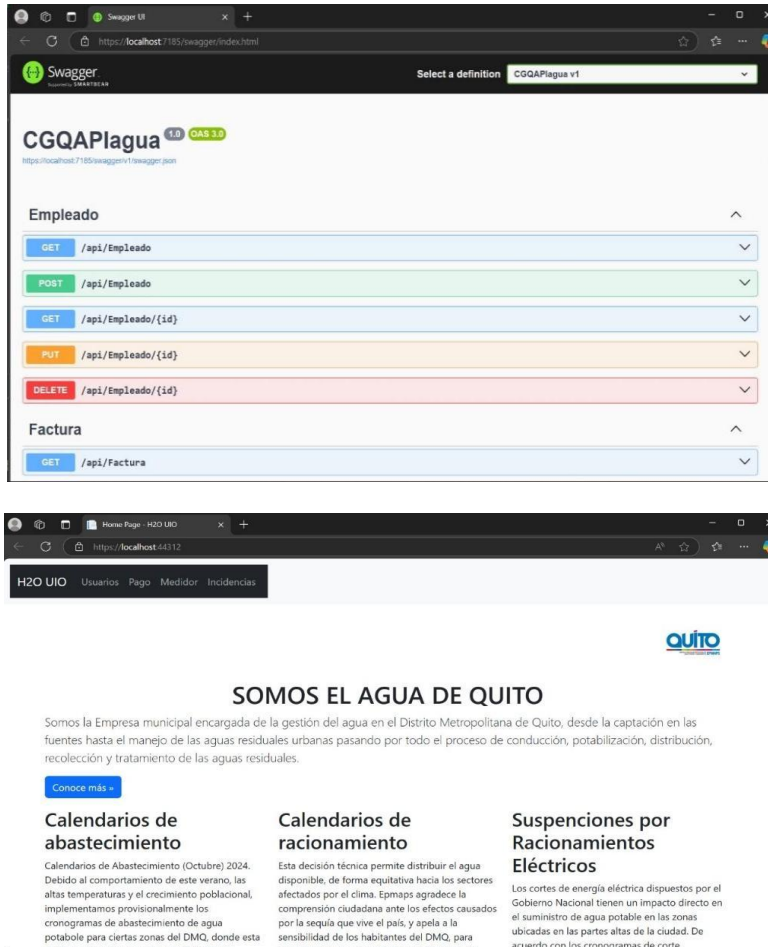
Aplicación Web API:

- **Propósito:** Permite que los datos gestionados por la aplicación MVC sean accesibles para aplicaciones externas como el frontend en .NET MAUI.
- **Tecnologías utilizadas:** Implementada con ASP.NET Core, expone endpoints RESTful para operaciones CRUD.
- **Seguridad:** Incluye autenticación y autorización mediante tokens JWT.
- **Estructura:** Los controladores de la API están organizados por entidades principales, como Usuarios, Medidas y Cobros.

Frontend

El frontend está desarrollado con .NET MAUI, proporcionando una aplicación multiplataforma que funciona tanto en dispositivos móviles como en escritorios.

- **Interfaz de Usuario:** Diseñada para ser intuitiva y fácil de usar, con menús claros que permiten a los usuarios navegar entre las funcionalidades principales.
- **Comunicación con la API:** Utiliza servicios HTTP integrados para realizar solicitudes a los endpoints de la API y mostrar los datos en tiempo real.
- **Base de Datos Local:**
 - Implementación de SQLite para permitir el acceso offline.
 - Los datos sincronizados incluyen usuarios, medidas de consumo y transacciones.



Requisitos del Sistema

Usuarios y Roles:

Administrador:

- Puede realizar cobros.
- Crear usuarios.
- Registrar medidas mensuales de consumo.

Cliente:

- Puede visualizar la base de datos sin realizar modificaciones.

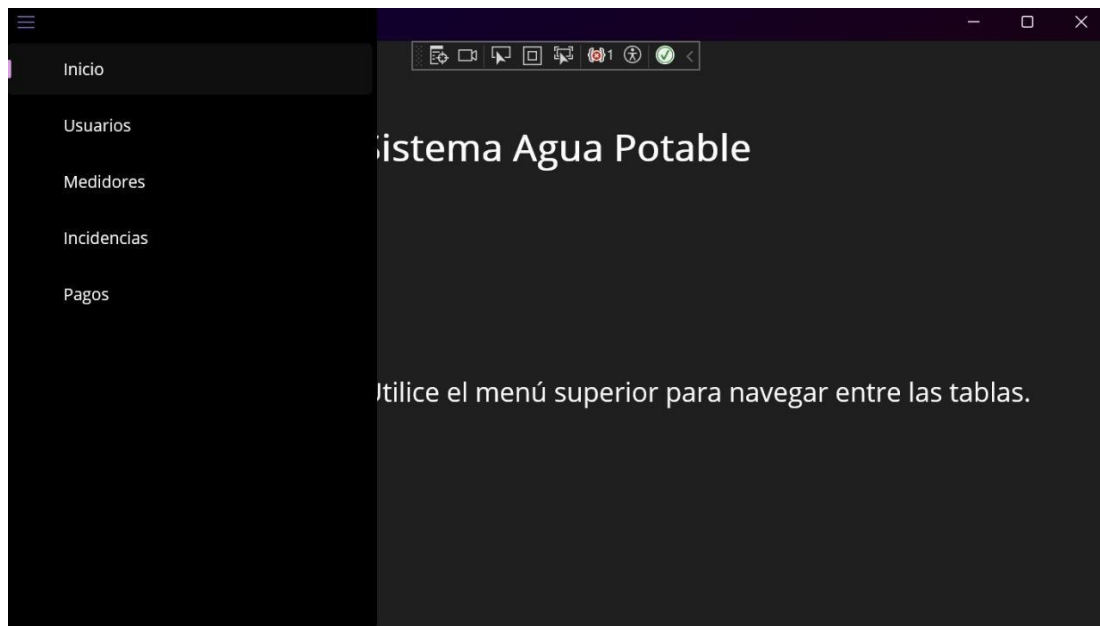
Funcionalidades Principales:

1. Login:

- Página de inicio donde los usuarios (administrador y cliente) inician sesión con su usuario y contraseña.

2. Página Principal:

- Acceso a las siguientes opciones:
- Cobrar.
- Crear usuario.
- Ver usuarios y consumo mensual.
- Añadir medida de consumo del mes actual.



Base de Datos

Registro de Usuarios:

- Almacena roles y permisos.

Registro de Medidas Mensuales:

- Almacena el consumo mensual de agua de cada usuario.

Historial de Cobros y Pagos:

- Registra todas las transacciones realizadas.

Base de Datos Local (SQLite):

- Permite almacenar temporalmente datos para funcionamiento offline.

Mejoras Futuras

Métodos de Pago Electrónicos:

Pagos mediante tarjetas, transferencias y billeteras digitales.

Predicción de Consumos:

Uso de IA para anticipar consumos futuros.

Notificaciones:

Alertas para pagos vencidos y consumos excesivos.

Gráficos Interactivos:

Reportes dinámicos sobre consumo mensual.

Conclusión

El sistema propuesto es una solución integral para la gestión del agua potable, optimizando los procesos administrativos y mejorando la experiencia de los usuarios. Su arquitectura modular y las nuevas implementaciones, como el patrón MVVM y las bases de datos locales, aseguran un funcionamiento eficiente y sostenible.

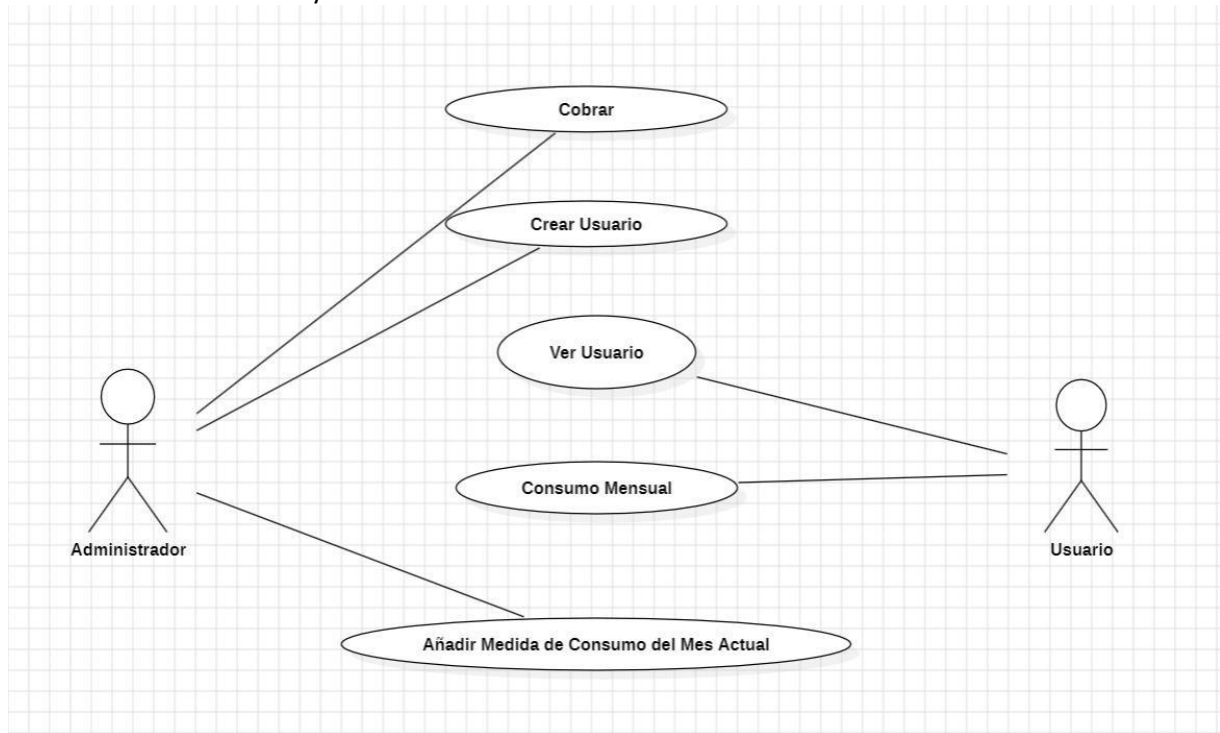


Figura 1: Caso de uso del sistema