Imagen que contiene Diagrama

Descripción generada automáticamente

ESTACION CLIMATOLOGICA

PROYECTO FERIA CIENTIFICA JUNIO 2024

ROLANDO ROQUE

LESLIE URBINA

JORGE RIVERA

FERNADO CHAVEZ

ENRIQUE COLINDRES

Contenido

[DESARROLLO DE UNA APLICACIÓN PARA EL MONITOREO CLIMÁTICO UTILIZANDO VISUAL STUDIO .NET 8, C#, C++, ARDUINO Y MYSQL 3](#_Toc170025162)

[INTRODUCCIÓN 3](#_Toc170025163)

[ANTECEDENTES 3](#_Toc170025164)

[JUSTIFICACIÓN 4](#_Toc170025165)

[PLANTEAMIENTO DEL PROBLEMA 4](#_Toc170025166)

[OBJETIVOS 5](#_Toc170025167)

[METODOLOGÍA 5](#_Toc170025168)

[RECURSOS 7](#_Toc170025169)

[MARCO TEÓRICO 8](#_Toc170025170)

[Visual Studio .NET 8 8](#_Toc170025171)

[Arduino 8](#_Toc170025172)

[MySQL 8](#_Toc170025173)

[ANÁLISIS Y DISEÑO 9](#_Toc170025174)

[Diagrama de Casos de Uso (Interfaz Usuario) 9](#_Toc170025175)

[Diagrama de Secuencias 9](#_Toc170025176)

[Diseño de Datos 10](#_Toc170025177)

[Base de Datos MySQL 10](#_Toc170025178)

[BIBLIOGRAFÍA 11](#_Toc170025179)

# DESARROLLO DE UNA APLICACIÓN PARA EL MONITOREO CLIMÁTICO UTILIZANDO VISUAL STUDIO .NET 8, C#, C++, ARDUINO Y MYSQL

Rolando Roque  
Leslie Urbina  
Jorge Rivera  
 Fernando Chávez  
 Enrique Colindres

# INTRODUCCIÓN

Este proyecto se enmarca en la creación de una aplicación para el monitoreo climático que integra diversas tecnologías de vanguardia. La aplicación está desarrollada utilizando Visual Studio .NET 8, C#, C++, Arduino y MySQL, con el objetivo de proporcionar una herramienta robusta y eficiente para la recolección y análisis de datos climáticos en tiempo real.

El monitoreo climático es esencial para múltiples disciplinas, tales como la agricultura, la gestión ambiental y la investigación científica. La recolección de datos climáticos se ha realizado tradicionalmente mediante métodos manuales o con sistemas aislados, limitando la capacidad para integrar y analizar la información de manera eficiente. Con el avance de la tecnología, es posible implementar sistemas automatizados que utilicen sensores y plataformas de desarrollo modernas para mejorar la precisión y la capacidad de análisis.

# ANTECEDENTES

Históricamente, el monitoreo climático ha sido una tarea que requiere mucho tiempo y esfuerzo humano. La recopilación manual de datos, como la temperatura y la presión atmosférica, no solo es propensa a errores, sino que también carece de la capacidad de proporcionar datos en tiempo real. Los sistemas existentes a menudo no ofrecen una solución integrada que combine tanto hardware como software de manera eficiente.

En este contexto, la utilización de la plataforma Arduino para la recolección de datos, combinada con el poder de procesamiento de lenguajes de programación como C# y C++ en Visual Studio .NET 8, y el manejo de datos mediante MySQL, representa un avance significativo en el desarrollo de soluciones tecnológicas para el monitoreo climático. Este proyecto pretende llenar el vacío existente, ofreciendo una solución integrada que combina hardware y software para la recolección, almacenamiento y análisis de datos en tiempo real.

# JUSTIFICACIÓN

La necesidad de un sistema de monitoreo climático automatizado y preciso es cada vez más evidente, especialmente en un mundo donde los cambios climáticos tienen un impacto significativo en diversas actividades humanas. Este proyecto pretende llenar ese vacío, ofreciendo una solución integrada que combina hardware y software para la recolección, almacenamiento y análisis de datos en tiempo real.

Visual Studio .NET 8 proporciona un entorno de desarrollo integrado (IDE) altamente eficiente, permitiendo el desarrollo de aplicaciones robustas y escalables. C# y C++ son lenguajes de programación ideales para el desarrollo de aplicaciones que requieren un alto rendimiento y una interfaz de usuario intuitiva. Arduino, por otro lado, es una plataforma de hardware flexible y de bajo costo, ideal para la recolección de datos en entornos diversos. MySQL, un sistema de gestión de bases de datos relacional, garantiza el almacenamiento seguro y eficiente de grandes volúmenes de datos.

# PLANTEAMIENTO DEL PROBLEMA

El monitoreo climático actual enfrenta varias limitaciones, incluyendo la falta de integración de sistemas y la dependencia de métodos manuales para la recolección de datos. Estos métodos no solo son propensos a errores humanos, sino que también carecen de la capacidad de proporcionar datos en tiempo real. Esto dificulta la toma de decisiones informadas y oportunas basadas en datos climáticos precisos.

El problema central que este proyecto aborda es la creación de un sistema automatizado que combine la recolección, almacenamiento y análisis de datos climáticos en una única plataforma. Este sistema debe ser capaz de operar en tiempo real, proporcionando datos precisos y actualizados que puedan ser utilizados para diversos fines, desde la investigación científica hasta la planificación agrícola y la gestión ambiental.

# OBJETIVOS

**Objetivo General**

Desarrollar una aplicación integral para el monitoreo climático utilizando Visual Studio .NET 8, C#, C++, Arduino y MySQL, que permita la recolección, almacenamiento y análisis de datos climáticos en tiempo real de manera eficiente y precisa.

**Objetivos Específicos**

1. **Implementación de Hardware**: Diseñar e implementar un sistema de recolección de datos climáticos utilizando sensores conectados a una plataforma Arduino.
2. **Desarrollo de Software**: Crear una interfaz de usuario intuitiva en C# y C++ para la visualización y análisis de los datos recolectados, utilizando Visual Studio .NET 8.
3. **Gestión de Base de Datos**: Configurar y administrar una base de datos MySQL para el almacenamiento eficiente de los datos climáticos.
4. **Pruebas y Validación**: Realizar pruebas exhaustivas del sistema para asegurar su precisión, fiabilidad y robustez, incluyendo pruebas unitarias, de integración y de aceptación por parte de los usuarios.

# METODOLOGÍA

**Metodología de Trabajo**

Se empleará una metodología CASCADA para el desarrollo de este proyecto, permitiendo la entrega de módulos funcionales en fases.

**Etapas del Modelo Cascada**

1. **Análisis**:
   * Recolección de requisitos mediante entrevistas con expertos en meteorología y análisis de sistemas de monitoreo existentes.
   * Definición de las especificaciones técnicas y funcionales del sistema, incluyendo los parámetros a monitorear y los sensores necesarios.
   * Diseño de la arquitectura del sistema, incluyendo diagramas de casos de uso y diagramas de flujo de datos.
2. **Diseño**:
   * Diseño de la base de datos MySQL, especificando las entidades y sus relaciones.
   * Diseño de la interfaz de usuario, incluyendo prototipos de pantalla y diagramas de navegación.
3. **Desarrollo**:
   * Implementación del código en Visual Studio .NET 8 utilizando C# y C++, asegurando la integración con la plataforma Arduino para la recolección de datos.
   * Desarrollo de la lógica de negocio y las capas de acceso a datos.
   * Configuración y programación de los sensores Arduino para la recolección de datos climáticos.
4. **Pruebas**:
   * Realización de pruebas unitarias para cada módulo del sistema, asegurando su correcto funcionamiento individual.
   * Pruebas de integración para verificar la interacción entre los diferentes componentes del sistema.
   * Pruebas de aceptación con usuarios finales para garantizar que el sistema cumple con los requisitos y expectativas definidos.
5. **Implementación**:
   * Despliegue del sistema en un entorno real, incluyendo la configuración de los sensores y la instalación del software en las estaciones de trabajo.
   * Capacitación de los usuarios finales en el uso del sistema y la interpretación de los datos climáticos.
   * Evaluación continua del rendimiento del sistema y la implementación de mejoras basadas en el feedback recibido.

# RECURSOS

**Recursos Hardware**

* **PC**: x86 64 bits Procesador que cumpla con los requisitos mínimos de Windows 10
* **Arduino**: Placas Arduino Uno equipadas con sensores BMP280 para la medición de temperatura, humedad y presión atmosférica.

**Recursos Software**

* **Sistema Operativo**: Windows 10.
* **Entorno de Desarrollo**: Visual Studio .NET 8, utilizado para el desarrollo de aplicaciones en C# y C++.
* **Arduino IDE**: Para la programación y configuración de las placas Arduino.
* **MySQL**: Sistema de gestión de bases de datos relacional, utilizado para el almacenamiento y gestión de los datos climáticos.
* **MySQL Workbench**: Herramienta de administración y diseño de bases de datos MySQL.
* **Bibliotecas y Frameworks**: .NET Framework, Entity Framework y bibliotecas específicas de Arduino.

# MARCO TEÓRICO

## Visual Studio .NET 8

Visual Studio .NET 8 es un entorno de desarrollo integrado (IDE) de Microsoft que soporta múltiples lenguajes de programación, incluidos C# y C++. Ofrece herramientas avanzadas para el desarrollo, depuración y mantenimiento de aplicaciones robustas y escalables. Este entorno es particularmente adecuado para el desarrollo de aplicaciones que requieren una interfaz de usuario rica y capacidades de procesamiento avanzadas. Además, su integración con servicios en la nube y herramientas de colaboración facilita el desarrollo de proyectos complejos y distribuidos.

## Arduino

Arduino es una plataforma de hardware libre basada en una placa con un microcontrolador y un entorno de desarrollo integrado (IDE). Esta plataforma es ampliamente utilizada en la creación de prototipos y proyectos electrónicos debido a su flexibilidad y facilidad de uso. Los sensores BMP280 utilizados en este proyecto permiten la recolección precisa de datos climáticos, incluyendo temperatura, humedad y presión atmosférica. La programación de Arduino se realiza en un lenguaje de programación basado en Wiring, lo que facilita la integración con otras plataformas de desarrollo como Visual Studio .NET.

## MySQL

MySQL es un sistema de gestión de bases de datos relacional de código abierto, conocido por su alto rendimiento, fiabilidad y facilidad de uso. Es ampliamente utilizado en aplicaciones web y empresariales para el almacenamiento y gestión de grandes volúmenes de datos. En este proyecto, MySQL se utiliza para almacenar los datos recolectados por los sensores Arduino, permitiendo un acceso rápido y eficiente a la información para su análisis y visualización. La estructura de la base de datos incluye tablas para usuarios, lecturas climáticas y logs de actividades, garantizando una gestión ordenada y segura de los datos.

# ANÁLISIS Y DISEÑO

## Diagrama de Casos de Uso (Interfaz Usuario)

1. **Recolección de Datos Climáticos**:
   * **Actor**: Usuario
   * **Descripción**: El usuario inicia el proceso de recolección de datos climáticos mediante la interfaz de la aplicación.
   * **Precondiciones**: Los sensores deben estar conectados y configurados correctamente en la plataforma Arduino.
   * **Postcondiciones**: Los datos recolectados se almacenan en la base de datos MySQL.
2. **Visualización de Datos**:
   * **Actor**: Usuario
   * **Descripción**: El usuario puede visualizar los datos climáticos recolectados en una interfaz gráfica.
   * **Precondiciones**: Los datos deben estar disponibles en la base de datos.
   * **Postcondiciones**: Los datos se presentan en gráficos y tablas para su análisis.

## Diagrama de Secuencias

1. **Recolección de Datos**:
   * El usuario inicia la recolección.
   * El sistema Arduino envía datos al servidor.
   * Los datos se almacenan en MySQL.
2. **Visualización de Datos**:
   * El usuario solicita la visualización.
   * El sistema recupera datos de MySQL.
   * Los datos se presentan en la interfaz gráfica.

# Diseño de Datos

## Base de Datos MySQL

**Entidades y Atributos**:

* **tblusuarios**: Tabla que almacena los datos de los usuarios del sistema.
  + usuario\_id: Identificador único.
  + user: Nombre de usuario.
  + passwd: Contraseña del usuario.
* **tbllecturas**: Tabla que almacena las lecturas climáticas registradas.
  + lectura\_id: Identificador único.
  + usuario\_id: Identificador del usuario que registró la lectura.
  + fecha\_hora: Fecha y hora de la lectura.
  + temperatura: Temperatura registrada.
  + presion: Presión atmosférica registrada.
  + altitud: Altitud registrada.
* **tbllogs**: Tabla que almacena los logs de actividades del sistema.
  + log\_id: Identificador único.
  + usuario\_id: Identificador del usuario asociado con la actividad.
  + actividad\_id: Identificador de la actividad realizada.
  + fecha\_hora: Fecha y hora de la actividad.

El diseño de la base de datos está optimizado para asegurar la integridad y consistencia de los datos, utilizando claves primarias y foráneas para establecer relaciones entre las tablas.

La implementación de esta aplicación permitirá mejorar la precisión y eficiencia en la recolección de datos climáticos, facilitando la toma de decisiones informadas basadas en datos precisos y actualizados. La capacidad de almacenar y analizar grandes volúmenes de datos en tiempo real es crucial para diversas aplicaciones, desde la investigación científica hasta la gestión ambiental y la planificación agrícola.

Este software es escalable a uso industrial, domiciliar, uso informativo en aplicaciones móviles.

# BIBLIOGRAFÍA

* Deitel y Deitel Como Programar en C++ PDF
* Shaums Outlines Series C++ PDF
* Videos y Manuales de Arduino.
* Arduino for Dummies PDF
* Videos de youtube
* Ingenieria de Software Roger Pressman PDF
* .NET 8 Manual del Programador Plataforma Microsoft Developer