**Laboratorio Virtual para Prácticas Académicas con IoT**

**1. Introducción**

**1.1. Contexto y Necesidad**

En el entorno académico, la formación práctica es fundamental para la comprensión y aplicación de conceptos teóricos. Los laboratorios tradicionales, aunque valiosos, tienen limitaciones en términos de flexibilidad y accesibilidad para la realización de prácticas en tiempo real. La creciente importancia de la tecnología y el Internet de las Cosas (IoT) en diversas disciplinas ha resaltado la necesidad de herramientas y sistemas que permitan a los estudiantes interactuar con hardware físico y software de manera remota y dinámica.

La IoT, por sus siglas en inglés "Internet of Things", se refiere a la interconexión de dispositivos a través de internet, permitiendo la recolección y el intercambio de datos en tiempo real. Esto abre nuevas oportunidades en la educación, facilitando la creación de laboratorios virtuales que permiten a los estudiantes acceder y controlar equipos y sensores desde cualquier lugar, a cualquier hora. Este enfoque no solo optimiza los recursos, sino que también prepara a los estudiantes para un mundo cada vez más digital y conectado.

**1.2. Beneficios del Sistema**

* Flexibilidad: Los estudiantes pueden realizar prácticas en cualquier momento y lugar, sin la necesidad de estar físicamente presentes en un laboratorio.
* Accesibilidad: Permite a un mayor número de estudiantes acceder a equipos y recursos que, de otro modo, podrían estar limitados por el espacio y la disponibilidad.
* Actualización Constante: Los laboratorios virtuales pueden actualizarse y mejorarse continuamente sin la necesidad de reemplazar equipos físicos.
* Preparación para el Futuro: Familiariza a los estudiantes con tecnologías emergentes y les brinda experiencia práctica en la gestión y análisis de datos en tiempo real.
* Interactividad y Control: Los estudiantes tendrán la capacidad de desarrollar y probar sus propias lógicas y configuraciones, promoviendo un aprendizaje más profundo y aplicado.
* Evaluación y Seguimiento: La capacidad de registrar y analizar las actividades de los estudiantes facilita una evaluación detallada del desempeño y el progreso durante las prácticas.
* Flexibilidad en el Despliegue: El sistema es versátil en su despliegue, pudiendo operar localmente a través de Wi-Fi o en la nube, lo que permite una amplia gama de aplicaciones en diferentes contextos educativos.

**1.3. Descripción del Proyecto**

Este proyecto propone el desarrollo de un sistema de laboratorio innovador basado en el chip ESP32, que integra principios de IoT para permitir a los estudiantes realizar prácticas educativas en control y monitoreo de sistemas electrónicos. El sistema estará compuesto por varios módulos de hardware y software que permitirán a los estudiantes configurar y controlar dispositivos electrónicos a través de una interfaz web.

**1.4. Objetivo General**

Desarrollar un sistema de laboratorio que combine hardware y software, utilizando el chip ESP32 y una interfaz web basada en Angular, para permitir a los estudiantes realizar prácticas educativas en control y monitoreo de sistemas electrónicos. Este sistema debe facilitar la configuración, control y evaluación remota de prácticas mediante el concepto de Internet de las Cosas (IoT).

**1.5. Objetivos Específicos**

1. **Diseñar e implementar el hardware necesario:**
   * Configurar el ESP32 como controlador principal.
   * Integrar sensores y actuadores adecuados para las prácticas educativas.
   * Desarrollar un tablero de conexiones para la manipulación física del sistema.
2. **Desarrollar una API Backend robusta:**
   * Crear una API que permita la comunicación eficiente entre el frontend, el ESP32 y la base de datos.
   * Asegurar que la API soporte las operaciones necesarias para el control y monitoreo de los dispositivos.
3. **Crear una interfaz de usuario intuitiva y funcional:**
   * Desarrollar una interfaz web utilizando Angular que permita a los estudiantes interactuar con el sistema de manera sencilla y efectiva.
   * Incluir funcionalidades que faciliten la configuración de sensores, actuadores y algoritmos de control.
4. **Implementar algoritmos de control en el ESP32:**
   * Programar el ESP32 para soportar salidas digitales, entradas digitales, entradas analógicas, y salidas analógicas.
   * Desarrollar algoritmos de control PID y lógica difusa que los estudiantes puedan utilizar y ajustar durante las prácticas.
5. **Garantizar la seguridad y privacidad de los datos:**
   * Implementar medidas de autenticación y autorización para los usuarios.
   * Asegurar la encriptación de datos tanto en tránsito como en reposo.
6. **Asegurar la escalabilidad del sistema:**
   * Diseñar la arquitectura del sistema para permitir la adición de nuevos módulos y sensores.
   * Facilitar la expansión tanto en hardware como en software según las necesidades educativas.
7. **Desarrollar herramientas de evaluación y seguimiento:**
   * Incluir funcionalidades que permitan registrar y analizar las actividades de los estudiantes.
   * Proveer métricas detalladas para evaluar el desempeño y progreso de los estudiantes.

**2. Requerimientos del Sistema**

**2.1. Requerimientos de Hardware**

* Chip ESP32 WROOM-32: Controlador principal para manejo de entradas y salidas digitales y analógicas.
* Tablero de Conexiones: Para evidenciar el funcionamiento del sistema de control con salidas y entradas cableadas.
* Sensores y Actuadores: Incluye sensores analógicos (como termocuplas) y actuadores para realizar las prácticas.

**2.2. Requerimientos de Software**

* Backend API: Para gestionar la comunicación entre el frontend, el ESP32 y la base de datos. Implementado en un framework como Spring Boot.
* Frontend: Interfaz de usuario desarrollada en Angular para la interacción remota con el ESP32.
* Configuración del ESP32: Módulos de programación y control para manejar salidas digitales, entradas digitales, entradas analógicas, salidas analógicas, y algoritmos de control PID y lógica difusa.

**3. Componentes del Sistema**

**3.1. Hardware**

* ESP32 W32-32: Procesador con capacidades de Wi-Fi para conexión remota y control de hardware.
* Tablero de Conexiones: Para visualizar y manipular las conexiones físicas, el tablero deberá facilitar la conexión de salidas digitales a módulos tipo relé que permita el manejo de cargas que requieren de altas corrientes. Cableado de entradas y salidas analógicas con el estándar de 4-20 ma.
* Sensores y Actuadores: Componentes para medir y controlar el entorno físico, como medidores de temperatura y nivel.

**3.2. Software**

* Módulo de Programación de Salidas Digitales: Funciones que permitan configurar las GPIO del ESP32 como salidas digitales.
* Módulo de Programación de Entradas Digitales: Funciones que permitan configurar GPIO del ESP32 como entradas digitales.
* Módulo de Programación de Entradas Analógicas: Funciones que permitan configurar y escalizar las entradas analógicas del ESP32.
* Módulo de Configuración de Controles PID y Lógica Difusa: Funciones que permitan ajustar y calibrar los algoritmos de control.
* Módulo de Gestión de Requisiciones HTTP: Funciones que permitan al ESP32 escuchar las requisiciones del backend y frontend.
* Desarrollo del Backend: Una API mediante la cual el ESP32 podrá acceder a una base de datos.
* Desarrollo del Frontend: Una interfaz visual mediante la cual el estudiante podrá configurar los sistemas de control y ajustar las variables de control.

**4. Capacidades del Sistema**

**4.1. Estudiantes Virtuales**

Los estudiantes que accedan de manera virtual podrán interactuar con el sistema a través de la interfaz web. Tendrán acceso a componentes y módulos pre-cableados y podrán diseñar sus propias lógicas para controlar y monitorear los sistemas.

**4.2. Estudiantes Presenciales**

Los estudiantes que se encuentren físicamente en el laboratorio podrán manipular el hardware directamente. Esto incluye la capacidad de cablear el sistema, configurar sensores y actuadores, y programar el ESP32 directamente. Esto permite una experiencia práctica más tradicional y complementa las capacidades virtuales.

**4.3. Seguridad y Privacidad**

Se implementarán medidas de seguridad para asegurar que los datos y el acceso al sistema estén protegidos. Esto incluye autenticación y autorización de usuarios, encriptación de datos en tránsito y en reposo, y políticas de acceso basadas en roles.

**4.4. Escalabilidad**

El sistema está diseñado para ser escalable, permitiendo agregar nuevos módulos y sensores a medida que se necesiten. La arquitectura permitirá la expansión tanto en hardware como en software, asegurando que el sistema pueda crecer con las necesidades educativas.

**4.5. Evaluación y Métricas**

El sistema incluirá herramientas para medir el desempeño y el progreso de los estudiantes. Esto incluye la capacidad de registrar y analizar las actividades de los estudiantes, facilitando una evaluación detallada y personalizada.

**5. Cronograma de Desarrollo e Implementación**

**5.1. Fase de Planificación (4 semanas)**

* Análisis de Requerimientos y Diseño: 2 semanas
* Planificación de Recursos y Cronograma: 2 semanas

**5.2. Fase de Desarrollo (12 semanas)**

* **Desarrollo del Hardware: 4 semanas**
  + Adquisición de Componentes
  + Montaje y Pruebas Iniciales
* **Desarrollo del Software Backend: 4 semanas**
  + Implementación de la API
  + Integración con la Base de Datos
* **Desarrollo del Software Frontend: 4 semanas**
  + Implementación de la Interfaz de Usuario
  + Integración con el Backend y el Hardware

**5.3. Fase de Pruebas (4 semanas)**

* **Pruebas de Integración y Funcionamiento: 2 semanas**
  + Verificación de la Comunicación entre Componentes
* **Pruebas de Usuario y Ajustes: 2 semanas**
  + Sesiones de Prueba con Estudiantes y Docentes
  + Ajustes Basados en Feedback

**5.4. Fase de Implementación (4 semanas)**

* Despliegue del Sistema en el Laboratorio: 2 semanas
  + Instalación y Configuración
* Configuración de Servicios en la Nube: 2 semanas
  + Pruebas de Acceso Remoto

**5.5. Fase de Capacitación y Documentación (4 semanas)**

* Capacitación a Docentes y Estudiantes: 2 semanas
  + Sesiones de Capacitación Presencial y Virtual
* Documentación del Sistema: 2 semanas
  + Manuales de Usuario y Guías de Referencia

**5.6. Mantenimiento y Actualización (Continuo)**

* Monitoreo del Sistema y Soporte Técnico:
  + Mantenimiento Preventivo y Correctivo
  + Actualización de Software y Hardware

**6. Conclusiones**

Este proyecto de Laboratorio Virtual para Prácticas Académicas con IoT no solo moderniza la forma en que los estudiantes interactúan con sistemas de control y monitoreo, sino que también prepara a los estudiantes para enfrentar los desafíos tecnológicos del futuro. La integración de tecnologías avanzadas como el ESP32 y la IoT en el entorno educativo abre nuevas oportunidades para el aprendizaje interactivo, accesible y flexible. La implementación de este sistema fortalecerá las capacidades prácticas de losestudiantes, facilitando una educación más completa y adaptada a las necesidades del siglo XXI.