[Fecha]

Iván Bezanilla López

IES Alisal

Sistema de Riego Automatizado con Arduino

Contenido

[PRESENTACIÓN 3](#_Toc165975519)

[CONTEXTO DEL PROYECTO 4](#_Toc165975520)

[OBJETIVOS 5](#_Toc165975521)

[Objetivos Principales: 5](#_Toc165975522)

[Objetivos Secundarios: 5](#_Toc165975523)

[ESQUEMA DE LA SOLUCIÓN 7](#_Toc165975524)

[Diagrama general: 7](#_Toc165975525)

[Diagrama de conexión de hardware: 9](#_Toc165975526)

[Diagrama de la estructura de AWS 12](#_Toc165975527)

[MATERIALES, RECURSOS y SERVICIOS NECESARIOS 15](#_Toc165975528)

[Hardware: 15](#_Toc165975529)

[Software: 19](#_Toc165975530)

[Servicios: 20](#_Toc165975531)

[PRESUPUESTO DETALLADO 21](#_Toc165975532)

[PLANIFICACIÓN DEL PROYECTO 22](#_Toc165975533)

[DESARROLLO TÉCNICO DEL PROYECTO 23](#_Toc165975534)

[Configuración del Hardware 23](#_Toc165975535)

[Ensamblaje y Cableado de los Componentes 23](#_Toc165975536)

[Configuración de la Fuente de Alimentación 23](#_Toc165975537)

[Desarrollo del Código para Arduino 24](#_Toc165975538)

[Verificación del Funcionamiento del Hardware 24](#_Toc165975539)

[Implementación en Entorno Local 24](#_Toc165975540)

[Preparación del Entorno de Desarrollo (Instalación de Ubuntu 22.04) 24](#_Toc165975541)

[Configuración del Servidor Web (Instalación de Apache, PHP y MySQL) 26](#_Toc165975542)

[Integración de MQTT con WebSocket (Instalación y Configuración de Mosquitto) 26](#_Toc165975543)

[Desarrollo de la Página Web y Base de Datos 27](#_Toc165975544)

[Pruebas de Conexión entre Arduino y la Página Web 27](#_Toc165975545)

[Migración a la Nube (AWS) 27](#_Toc165975546)

[Creación de una Infraestructura Virtual Privada (VPC) 27](#_Toc165975547)

[Configuración de Grupos de Seguridad 30](#_Toc165975548)

[Implementación de Instancias EC2 y Servicios Asociados 33](#_Toc165975549)

[Configuración de un Servicio de Base de Datos Relacional (RDS) 35](#_Toc165975550)

[Transición de la Página Web a la Nube (EC2) 35](#_Toc165975551)

[Migración de la Base de Datos a la Nube (RDS) 35](#_Toc165975552)

[Verificación y Validación 35](#_Toc165975553)

[Pruebas de Funcionamiento en el Entorno Local y en la Nube 35](#_Toc165975554)

[Evaluación de la Seguridad y el Rendimiento 35](#_Toc165975555)

[Confirmación del Éxito del Proyecto 35](#_Toc165975556)

[ANÁLISIS DE RIESGOS 36](#_Toc165975557)

[BIBLIOGRAFÍA 36](#_Toc165975558)

[ANEXOS 36](#_Toc165975559)

# PRESENTACIÓN

**GreenTech Solutions: Líder en Tecnología Verde para una Agricultura Sostenible**

En un mundo cada vez más consciente del medio ambiente, GreenTech Solutions se enorgullece en presentar nuestro último proyecto: el Sistema de Riego Automatizado con Arduino. Esta innovadora solución fusiona tecnología de vanguardia con nuestra pasión por la agricultura y la sostenibilidad, abriendo nuevas puertas hacia una forma más inteligente y eficiente de cuidar nuestras plantas.

Nuestro sistema se destaca por su capacidad para monitorear de manera precisa y continua las condiciones del suelo y del ambiente, gracias a la integración de múltiples sensores. Estos datos se almacenan y analizan en una base de datos MySQL, proporcionando información detallada que permite optimizar el riego y mejorar el crecimiento de las plantas.

Además, hemos desarrollado una interfaz web interactiva alojada en un servidor Apache, que brinda acceso en tiempo real a los datos de los sensores y permite el control remoto del sistema de riego desde cualquier lugar. Esta facilidad de acceso y control intuitivo se traduce en un cuidado más eficiente de las plantas, facilitando la vida tanto en hogares como en empresas agrícolas.

La comunicación entre el Arduino Uno y Apache se logra a través de MQTT, asegurando una transmisión rápida y confiable de datos. Otro aspecto destacado de nuestro sistema es la bomba de agua controlada por Arduino, que ofrece la flexibilidad de personalizar la duración y frecuencia del riego según las necesidades específicas de cada planta.

Los beneficios de nuestro Sistema de Riego Automatizado son abundantes. Desde la optimización del uso del agua hasta la reducción de costos y tiempo dedicado al riego manual, pasando por la contribución a la conservación del medio ambiente al evitar el desperdicio de agua y promover prácticas de riego más sostenibles, estamos comprometidos con un futuro más verde y próspero para todos.

# CONTEXTO DEL PROYECTO

El proyecto de desarrollo de un sistema de riego automatizado con Arduino se enmarca en la necesidad de encontrar soluciones innovadoras y sostenibles para optimizar el cuidado de las plantas en entornos tanto domésticos como empresariales.

Con la preocupación creciente por el uso eficiente del agua y la conservación del medio ambiente, GreenTech Solutions se ha embarcado en la misión de desarrollar una tecnología que no solo simplifique el proceso de riego, sino que también promueva prácticas agrícolas más responsables.

Este proyecto se llevará a cabo en dos escenarios distintos, cada uno adaptado a las necesidades y requisitos específicos de los usuarios involucrados: personas individuales y empresas agrícolas.

Para personas individuales, el sistema de riego automatizado ofrecerá una solución práctica y accesible para el cuidado de jardines, huertos o áreas verdes en hogares y comunidades residenciales. La interfaz intuitiva y el control remoto a través de dispositivos móviles permitirán a los usuarios monitorear y gestionar el riego de sus plantas de manera conveniente y eficiente, incluso cuando no estén en casa.

Por otro lado, en el contexto de empresas agrícolas, el sistema de riego automatizado se presenta como una herramienta indispensable para la optimización de procesos y la mejora de la productividad. Con la capacidad de adaptarse a diferentes tipos de cultivos y tamaños de terreno, esta tecnología permitirá a los agricultores controlar el riego de manera precisa y eficiente, maximizando el rendimiento de sus cosechas mientras minimizan el uso de recursos como el agua y la energía.

En ambos escenarios, el proyecto se basa en principios de innovación, eficiencia y sostenibilidad. GreenTech Solutions está comprometido con el desarrollo de soluciones tecnológicas que no solo mejoren la calidad de vida de las personas, sino que también contribuyan a la preservación del medio ambiente y la creación de un futuro más verde y próspero para todos.

# OBJETIVOS

## Objetivos Principales:

1. Desarrollar un sistema de riego automatizado con Arduino funcional para entornos individuales.
   * A corto plazo: Investigar y seleccionar los sensores adecuados para monitorear las condiciones del suelo y del ambiente.
   * A medio plazo: Diseñar e implementar el circuito electrónico que permita la comunicación entre los sensores, Arduino y la bomba de agua.
   * A largo plazo: Realizar pruebas integrales del sistema en entornos individuales, ajustar y optimizar el funcionamiento según los resultados obtenidos.
2. Implementar una interfaz web interactiva que muestre los valores de los sensores en tiempo real.
   * A corto plazo: Diseñar la estructura y la interfaz de usuario de la aplicación web.
   * A medio plazo: Desarrollar y conectar la interfaz web con la base de datos MySQL para almacenar y recuperar los datos de los sensores.
   * A largo plazo: Realizar pruebas de integración entre el sistema de riego y la interfaz web, asegurando la correcta visualización de los datos.

## Objetivos Secundarios:

1. Habilitar la funcionalidad desde la interfaz web para activar el riego durante los segundos indicados.
   * A corto plazo: Implementar la lógica de control en la interfaz web para permitir la activación remota del riego.
   * A medio plazo: Integrar la lógica de control con el sistema de riego automatizado, asegurando la sincronización y la precisión en la activación del riego.
   * A largo plazo: Realizar pruebas exhaustivas de la funcionalidad de activación remota del riego, corrigiendo posibles errores y mejorando la experiencia del usuario.
2. Adaptar el sistema de riego automatizado para entornos empresariales.
   * A corto plazo: Identificar las necesidades específicas de las empresas agrícolas y las características adicionales requeridas para el sistema.
   * A medio plazo: Desarrollar módulos adicionales o funcionalidades específicas para satisfacer los requisitos empresariales, como la gestión de múltiples zonas de riego.
   * A largo plazo: Realizar pruebas piloto del sistema adaptado en empresas agrícolas, recopilando comentarios y retroalimentación para realizar ajustes finales.
3. Diseñar un sistema de control inteligente que ajuste automáticamente el riego según las necesidades de las plantas.
   * A corto plazo: Investigar algoritmos de control y sistemas de reglas para la gestión eficiente del riego.
   * A medio plazo: Implementar el sistema de control inteligente en el software del Arduino, utilizando los datos recopilados por los sensores.
   * A largo plazo: Realizar pruebas de funcionamiento del sistema de control inteligente, evaluando su capacidad para adaptarse dinámicamente a las condiciones cambiantes del entorno.

En resumen, los objetivos principales del proyecto se centran en el desarrollo y la implementación exitosa del sistema de riego automatizado, mientras que los objetivos secundarios se enfocan en aspectos como la usabilidad, la eficiencia del riego y la adaptabilidad del sistema a las necesidades de los usuarios. Estos objetivos se han establecido con plazos claros a corto, medio y largo plazo para garantizar un progreso constante y medible a lo largo del proyecto.

# ESQUEMA DE LA SOLUCIÓN

## Diagrama general:

El presente documento tiene como objetivo realizar una detallada descripción del sistema de comunicación de datos presente en la imagen proporcionada, analizando sus componentes, flujos de información y funcionalidades.

**Componentes del sistema**

El sistema se compone de los siguientes elementos:

* **Arduino Uno:** Placa microcontroladora que sirve como núcleo del sistema, recopilando datos de los sensores y enviándolos al servidor MQTT.
* **Sensores:**
  + Sensor DHT11: Mide la temperatura y la humedad del aire.
  + Sensor de humedad del suelo: Mide la humedad del suelo.
* **Servidor MQTT:** Instancia EC2 en la nube de Amazon Web Services (AWS) que recibe los datos de los sensores, los procesa y los distribuye a otros componentes.
* **Apache:** Servidor web que aloja la aplicación web para visualizar los datos de los sensores.
* **PHP:** Lenguaje de programación que interactúa con la base de datos MySQL para almacenar los datos de los sensores.
* **MySQL:** Base de datos relacional que almacena los datos de los sensores en una tabla.
* **Interfaz web:** Página web que permite visualizar los datos de los sensores en tiempo real y establecer el tiempo de activación de la bomba de agua.
* **Bomba de agua:** Dispositivo que se activa en función del tiempo establecido en la interfaz web, controlando el riego de las plantas.

**Flujos de información**

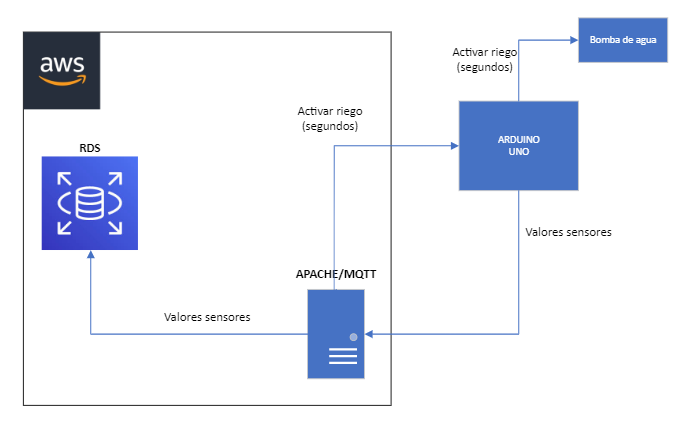
El sistema funciona mediante la interacción de sus componentes y el intercambio de información de la siguiente manera:

1. **Recolección de datos:** Los sensores DHT11 y de humedad del suelo envían sus datos al Arduino Uno.
2. **Envío a servidor MQTT:** El Arduino Uno envía los datos de los sensores al servidor MQTT mediante el protocolo MQTT.
3. **Procesamiento y distribución:** El servidor MQTT recibe los datos de los sensores, los procesa y los distribuye a Apache y a la interfaz web.
4. **Visualización en la web:** Apache recibe los datos procesados del servidor MQTT y los utiliza para generar la interfaz web que muestra los datos de los sensores en tiempo real.
5. **Almacenamiento en base de datos:** PHP interactúa con la base de datos MySQL para almacenar los datos de los sensores en una tabla específica.
6. **Control de la bomba de agua:** La interfaz web permite al usuario establecer un tiempo en segundos para la activación de la bomba de agua. El servidor MQTT recibe este tiempo y lo envía al Arduino Uno.
7. **Activación de la bomba:** El Arduino Uno recibe el tiempo de activación de la bomba de agua del servidor MQTT y activa la bomba durante el tiempo especificado.

**Funcionalidades del sistema**

El sistema de comunicación de datos presentado ofrece las siguientes funcionalidades:

* **Monitoreo remoto de datos de sensores:** Permite visualizar en tiempo real los valores de temperatura, humedad del aire y humedad del suelo desde una interfaz web.
* **Almacenamiento histórico de datos:** Almacena los datos de los sensores en una base de datos para su análisis posterior y la toma de decisiones informadas sobre el riego de las plantas.
* **Control remoto de la bomba de agua:** Permite al usuario establecer un tiempo específico para la activación de la bomba de agua, automatizando el proceso de riego.
* **Escalabilidad:** El sistema puede ampliarse fácilmente para incluir más sensores y dispositivos, adaptándose a las necesidades específicas de cada proyecto.



## Diagrama de conexión de hardware:

Este diagrama detallará cómo se conectan físicamente los componentes del hardware.

Mostrará cómo se conectan los sensores (DHT11, sensor de humedad del suelo) al Arduino Uno, cómo se conecta el módulo ESP8266-01S al Arduino Uno y cómo se conecta la mini bomba de agua al relé conectado al Arduino Uno.

El presente documento tiene como objetivo realizar una detallada descripción del circuito electrónico presente en la imagen proporcionada, analizando sus componentes, conexiones, funcionamiento, aplicaciones, posibles mejoras y recursos adicionales.

**Componentes**

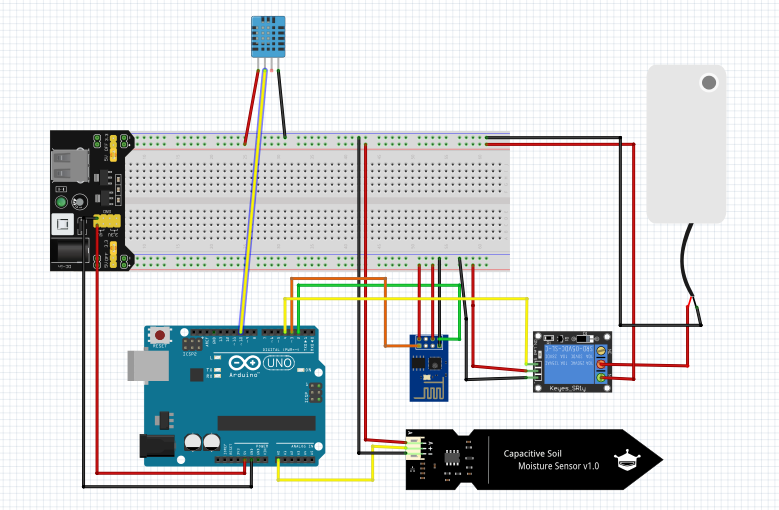
El circuito se compone de los siguientes elementos:

* **Arduino Uno:** Placa microcontroladora que sirve como núcleo del sistema, procesando y ejecutando las instrucciones del programa.
* **Sensor DHT11:** Sensor ambiental que mide la temperatura y la humedad del aire, proporcionando datos valiosos para el control del sistema.
* **Sensor de humedad del suelo:** Sensor especializado en medir la humedad del suelo, permitiendo determinar la necesidad de riego para las plantas.
* **ESP-01:** Módulo WiFi que posibilita la conexión del circuito a internet, enviando los datos de los sensores a un servidor web para su visualización y análisis.
* **Relé:** Componente electrónico que actúa como interruptor controlado eléctricamente, permitiendo encender y apagar la bomba de agua.
* **Bomba de agua:** Dispositivo que impulsa el agua para el riego de las plantas, activada por el relé.
* **MB-102:** Módulo de alimentación.

**Conexiones**

Las conexiones entre los componentes se establecen de la siguiente manera:

* **Sensor DHT11:**
  + VCC a 3.3V del Arduino: Proporciona alimentación al sensor.
  + GND a GND del Arduino: Establece la conexión a tierra para la referencia de voltaje.
  + DATA a pin 10 del Arduino: Transfiere los datos de temperatura y humedad al Arduino.
* **Sensor de humedad del suelo:**
  + VCC a 5V del Arduino: Suministra alimentación al sensor.
  + GND a GND del Arduino: Referencia de voltaje para las mediciones.
  + DATA a pin A0 del Arduino: Envía los valores de humedad del suelo al Arduino.
* **ESP-01:**
  + VCC a 3.3V del Arduino: Alimentación para el módulo WiFi.
  + GND a GND del Arduino: Referencia de tierra.
  + CH\_PD a 3.3V del Arduino: Enciende y apaga el módulo WiFi.
  + TX a pin 2 del Arduino: Envía datos desde el ESP-01 al Arduino.
  + RX a pin 3 del Arduino: Recibe datos del Arduino al ESP-01.
* **Relé:**
  + VCC a 5V del Arduino: Alimentación para el relé.
  + GND a GND del Arduino: Referencia de tierra.
  + IN1 a pin 9 del Arduino: Recibe la señal de control del Arduino para activar o desactivar el relé.
  + COM a bomba de agua: Conecta el relé a la bomba de agua.
* **Bomba de agua:**
  + COM del relé: Conexión positiva de la alimentación de la bomba.
  + GND del Arduino: Conexión negativa de la alimentación de la bomba.
* **MB-102:**
  + VCC a 5V del Arduino: Proporciona alimentación al módulo Bluetooth.
  + GND a GND del Arduino: Referencia de tierra.



## Diagrama de la estructura de AWS

En esta presentación, se explicará el diseño de una red en AWS que utiliza algunos de los servicios de red de AWS.

**Componentes principales**

La red se compone de los siguientes componentes principales:

* **Amazon Virtual Private Cloud (VPC)**: Una VPC es una red privada virtual que te permite aislar tus recursos de AWS de la red pública de Internet. Esto te ayuda a proteger tus recursos de accesos no autorizados.
* **Subredes**: Las subredes son divisiones lógicas de una VPC. Puedes utilizar subredes para segmentar tu red y controlar el tráfico entre diferentes tipos de recursos.
* **Instancias EC2**: Las instancias EC2 son servidores virtuales que puedes ejecutar en la nube de AWS. Puedes utilizar instancias EC2 para ejecutar aplicaciones, almacenar datos y mucho más.
* **Amazon RDS**: Amazon RDS es un servicio de base de datos administrado que te permite ejecutar bases de datos relacionales en la nube de AWS. Amazon RDS se encarga de la administración de la base de datos, como las actualizaciones de software, las copias de seguridad y la protección contra fallos.
* **Puerta de enlace de Internet**: Una puerta de enlace de Internet es un dispositivo que te permite conectar tu VPC a Internet. Esto te permite que tus recursos de AWS sean accesibles desde Internet.

**Flujo de tráfico**

El flujo de tráfico en la red se puede describir de la siguiente manera:

* Los usuarios de Internet pueden acceder a la base de datos RDS a través de la dirección IP pública de la subred pública. Esto permite que los usuarios de Internet se conecten a la base de datos y ejecuten consultas.
* Las instancias EC2 pueden acceder a la base de datos RDS a través de la dirección IP privada de la base de datos. Esto permite que las instancias EC2 se conecten a la base de datos y ejecuten consultas.
* Las instancias EC2 pueden acceder entre sí a través de las direcciones IP privadas dentro de la VPC. Esto permite que las instancias EC2 se comuniquen entre sí sin tener que salir a Internet.

**Consideraciones de seguridad**

La arquitectura de red de la imagen incluye algunas medidas de seguridad básicas, como:

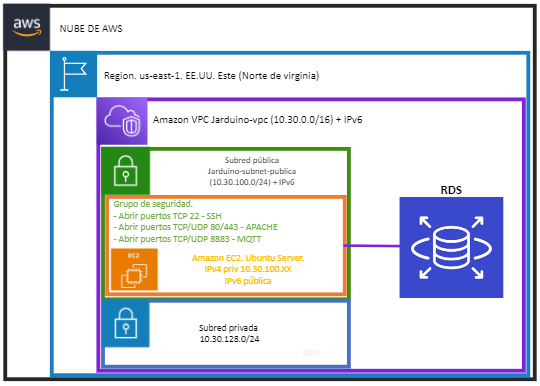
* **Grupo de seguridad**: El grupo de seguridad de la instancia EC2 permite el acceso a los puertos que son necesarios para que la instancia funcione.
* **Subredes privadas**: Los recursos de la zona de red privada no están directamente expuestos a Internet.

Sin embargo, la arquitectura podría mejorarse con la implementación de medidas de seguridad adicionales, como:

* **Lista de control de acceso (ACL)**: Una ACL es una lista de reglas que permiten o deniegan el tráfico de red a una instancia EC2. Puedes utilizar ACL para restringir aún más el acceso a los puertos de la instancia EC2.
* **Cifrado de datos**: Los datos en reposo y en tránsito deben estar cifrados para protegerlos de accesos no autorizados.

**Conclusión**

El diseño de red que se muestra en la imagen es una buena base para una red en AWS. Sin embargo, se pueden realizar algunas mejoras para mejorar la seguridad de la red.



# MATERIALES, RECURSOS y SERVICIOS NECESARIOS

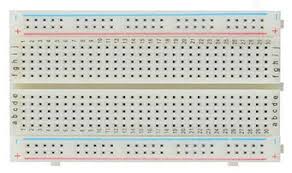
## Hardware:

**1. Arduino Uno R3:**

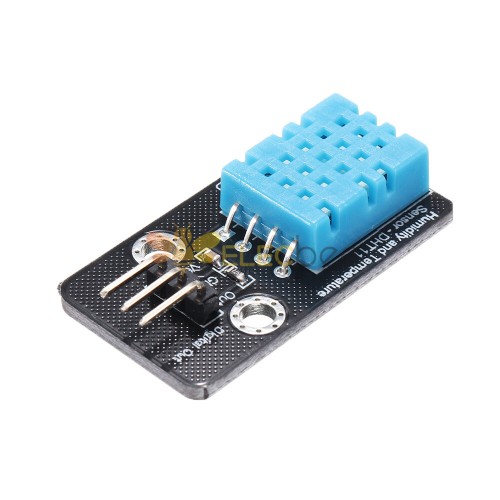
* + Microcontrolador ATmega328.
  + Características: Puerto USB, 14 pines digitales de entrada/salida (6 de ellos pueden ser PWM), 6 entradas analógicas, velocidad de reloj de 16 MHz.
  + Precio: 22,99 € (IVA incluido)
  + [Enlace](https://www.amazon.es/Arduino-UNO-A000066-microcontrolador-ATmega328/dp/B008GRTSV6/ref=sr_1_5?crid=3QBQVS22GUC7A&dib=eyJ2IjoiMSJ9.N8nF-4nrSGrUBX0eJZ8DQ3faJEQ6oU7jDfwMD55RC6PHY2xj3N9JePg85PFtW5v-nG1TLbyHvS9oNR5mPbB6sJNW75nQi8cvNy4kP0iT2t9zmy-F27ZnrA6bcqa8AZZNar1W1C-38gvaebG-VIpUeLuLxk2SnyFVRtfiXKLinuQd5iS-ovK5JxbNKNFDLE2FR01vFOMqDdFPUTTZQ6u4iDywcYGhDjz4-p_AGgxXiMcNd0uKmJsMRmqdGunroAgjvu3sF0CX8juMP2dBEHezR08bpnT6785feExFtGKqQeU.Xk6fvyzP2I-TnIgTqIMWK7zQALHMukrGGGX1qA2FSmo&dib_tag=se&keywords=arduino+uno+r3&qid=1713425949&sprefix=arduino+%2Caps%2C102&sr=8-5)

**2. Placa de prueba:**

* + Incluye barras conductoras para facilitar la conexión de componentes.
  + Dimensiones: 16.5 cm x 5.5 cm.
  + Precio: 5,49 € (IVA incluido)
  + [Enlace](https://www.amazon.es/AZDelivery-Breadboard-barras-conductoras-Arduino/dp/B07KKJSFM1/ref=sr_1_5?crid=2F0BISXJE3DXY&dib=eyJ2IjoiMSJ9.qBPp7lHWWelJcqFJQYHaU0QuRzF7SqCy4PDW4Vus6H3i6FWCMCrJcUbWfocvBWiSrcpqZsns_aj-tGxVm46qMedXsErejA0wjcfdOf1kCLyJ2JrjAkkNHAGPXJnBufM0rvlByYZu-z--8kQSHVK2sqW7Pl0UG4Y7npkQe6TopjizWjn-UbA2-4b2jBGyobo__nowEZfJylMCyXCy5eWudUgpD3jMlgkeemq3v0SgJNp5GFHzYSsZk8ThjJNGFy9rRxNO2spD0ZbTRmkybW_gMfZychMC0zJ6p-fOY6aJTSI.JJS4vaJrBGw3FZS3Cp4WbMK3sYPlUuD8YwZCV7fqLIg&dib_tag=se&keywords=placa+de+pruebas+arduino&qid=1713428538&sprefix=placa+de+pruebas+ar%2Caps%2C103&sr=8-5)



**3. Sensor DHT11:**

* + Sensor digital de temperatura y humedad.
  + Rango de temperatura: 0°C a 50°C.
  + Rango de humedad: 20% a 90% HR.
  + Precio: 5,99 € (IVA incluido)
  + [Enlace](https://www.amazon.es/AZDelivery-Temperatura-Compatible-Arduino-Incluido/dp/B089W8DB5P/ref=sr_1_1_sspa?dib=eyJ2IjoiMSJ9.J8_7o_7nkjidzjJwsOQSVGJTk_DI7naVgLCPtovUVJ7ObflO0e2acy4CTzlM1qZjY0EkKvQ3zgaUXruE6kQBgGJ88zYS0NGs-7J0HgddAVIUDNs8NCl-rW76F0seUGP0QFqYYo8JFtAgv5lILF93f1ZU2sPjxq6MfthSiuoTU9ndeVl0DgVSUwDLCXc5q7lzvQo6QVhoXGwiRi3qs5-6Yd-BooWL4U1QcterRJG73n3VAkXL0sxmcViU5cHgBCCbTmR61kVS9GCb1kN1u747vpsdhJK65mYaTMc6M_XHMho.GTL45kmFU2Jy-XPbw0l0qWKS8N4lllHWz6roPgCiqD4&dib_tag=se&keywords=dht11&qid=1713425766&sr=8-1-spons&sp_csd=d2lkZ2V0TmFtZT1zcF9hdGY&th=1)

**4. Sensor de humedad del suelo:**

* + Módulo higrómetro V1.2.
  + Voltaje de operación: 3.3V - 5V.
  + Salida analógica.
  + Precio: 7,99 € (IVA incluido)
  + [Enlace](https://www.amazon.es/AZDelivery-Sensor-Humedad-Suelo-Modulo_higrometro_V1-2_Parent/dp/B07YY2RN1B)

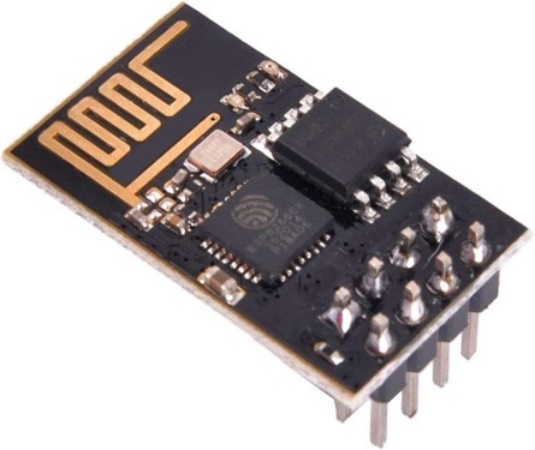
**5. Relé:**

* + Módulo KY-019 con relé de un canal.
  + Capacidad de conmutación: 250VAC/10A, 125VAC/10A, 30VDC/10A.
  + Precio: 5,99 € (IVA incluido)
  + [Enlace](https://www.amazon.es/ARCELI-KY-019-M%C3%B3dulo-Shield-arduino/dp/B07BVXT1ZK/ref=sr_1_5?__mk_es_ES=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=24W9AAL2CKL&dib=eyJ2IjoiMSJ9.aLr9rUAXO6XYld128fwZUOdseJrgSo9EAxvVFD_K-nFhk00N-3LbBSYSuUje4iuGtdhRQoEyxGQiDVbscf2wGVuJ3yAq_iY_EryjJ5TLalakasEeVDkrV-xpzMeI-oqcOnuxC8Ht8VkP4gqMpf2H_rSNLWO-GoDRp6tPvrII9WD4GDkitzA8bEglqU7wiBYLGTg9ytEVr-fjD1b0bSAfUwKcCuBYVNym5MIQH0Gxy1hPxchIhvcAhRFToRfigwripqTth0ZZRHf-9gPuo5LbFD5xTL4KiIkjbOVoZQeztoQ.svSW2iXIKFmMxjKUAOkymVg7f7iUjMCoQftV6Og2Zvg&dib_tag=se&keywords=rele+arduino&qid=1713425822&sprefix=rele+arduino%2Caps%2C98&sr=8-5)

**6. Mini bomba de agua:**

* + Bomba sumergible con cepillo y tubería para tanque.
  + Voltaje de operación: 3V - 6V.
  + Caudal máximo: 100L/h.
  + Altura máxima de elevación: 40-110cm.
  + Precio: 8,99 € (IVA incluido)
  + [Enlace](https://www.amazon.es/RUNCCI-YUN-Sumergible-Cepillo-Tuber%C3%ADa-paraTanque/dp/B082PM8L6X/ref=sr_1_5?crid=2MUPU3MCNTPQT&dib=eyJ2IjoiMSJ9.BvDnZOOEMe1jbq75fCUEGvu5WhCMollyOuhoSvwHUH2dMISFgI5-Tvh-N20KiBdzR6dcPa12N2VYjiuGCN0jt66Qc2e1J25ESUUNB4InJIQ-skXD7js1nCVtwmA25LruAR6k4HhvuFCGUoV2t_zh7XEpbr06zvCP6qfXkkhCaJuefjSL3wcWu6T2hjXErlVuPX97OvdpH8wM9M_AF3blwQsPwG44LApOT7bFTFkANZKut0WdXX3hcxv_YjzKC7NPsPK-Q5WYUK2xJkGcIhsRIeG0_Ewq2VWhKEP6pa94zOQ.fn2JSNYRdVeODKQ7FoVtuywYHlXZ8xu4gH3lW4BjWn0&dib_tag=se&keywords=bomba%2Bde%2Bagua%2Bmini%2Barduino&qid=1713425854&sprefix=mini%2Bbomba%2Bde%2B%2Barduino%2Caps%2C92&sr=8-5&th=1)

**7. Módulo ESP8266-01:**

* + Adaptador para breadboard del módulo ESP8266-01.
  + Voltaje de operación: 3.3V.
  + Compatible con la mayoría de los microcontroladores.
  + Precio: 6,49 € (IVA incluido)
  + [Enlace](https://www.amazon.es/AZDelivery-ESP8266-01S-Adaptador-Breadboard-Compatible/dp/B072R6DPK7/ref=sr_1_3?__mk_es_ES=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=1TAMWWNELPA1J&dib=eyJ2IjoiMSJ9.cc3weZljGH3eErmWCpTXT6inbUpVGXjTMvHCPQDanlVyv-lfEfQKtVHB-mWQ3027PwhbugX3tNSZ075GH8jqVoU_kHCSMI0-ubGB-67zyf-iE8JAV_sJnEWS53tfkLHbGqg8VJywmuCpArCUbKxUB7HRdH1OO6a_6zTFO4-xjCmpCH4LVB4kZO-J4egCkphwzgnmJRbW75CDBNCBvSx6Hr80x8cbCesCii2XyeH_AjSlHA8iKsVZDdDhoRb4yLzAfMpUUNlO_LzLoUuRIdzdl3SIaUh36PBz2dC7IsdONOg.i7E123vzvJIdndpLMyY3_Lm74n9o0NPVtW7b41dVWVg&dib_tag=se&keywords=modulo+esp8266-01&qid=1713428466&sprefix=modulo+esp8266-01%2Caps%2C108&sr=8-3)

**8. Módulo de alimentación:**

* + Adaptador de corriente para breadboard.
  + Voltaje de entrada: 6.5V - 12V.
  + Voltaje de salida: 3.3V / 5V seleccionable.
  + Precio: 4,99 € (IVA incluido)
  + [Enlace](https://www.amazon.es/AZDelivery-MB102-Breadboard-Adaptador-alimentaci%C3%B3n/dp/B06X962SPW/ref=sr_1_5?__mk_es_ES=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=TSPVR1GDST12&dib=eyJ2IjoiMSJ9.zJGROkc0oHv54fSXGaAVPepFRb91MUsusVPsob4hadntwTtQgZuOzDh8qfa6Ggd7hxloq8lVl0pj0zP-aW51rBawtr2B8TgxObb38zf8IrMD3C8czRgO1wGbOARpaWxgG2BOvNBWaWAFfe-PS1NXwPi0_2xnHcSpiivieT8VaRvkcWByIiyCc-A-eF1yj9pVOPPxrkgAW-wPKh3xjR8RoGua96pFZqjdvYcJK3KMAoR_Q6SaAiCo8HG-rcUN4MZ7VFTK59QxMb-VxXxGJ5BH1Nl1N4DpBc9FsqaJwfABzG8.c6FgKmMW3m1zpCflNlnzsKlAhCgIWX8bDXYevGGr4Fs&dib_tag=se&keywords=modulo%2Bfuente%2Bde%2Balimentacion%2Barduino&qid=1713428394&sprefix=modulo%2Bfuente%2Bde%2Balimentacion%2Barduino%2Caps%2C97&sr=8-5&th=1)

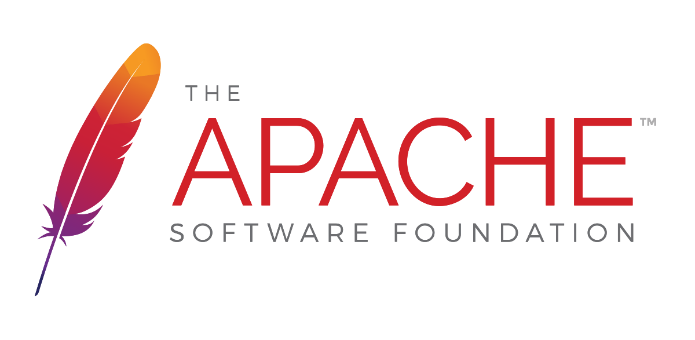


## Software:

**1. RDS MySQL:**

* + Sistema de gestión de bases de datos relacionales.
  + Versión: MySQL 8.0.
  + Precio: Variable dependiendo del plan de AWS.

**2. Apache:**

* + Servidor web HTTP de código abierto.
  + Versión: Apache 2.4.
  + Precio: Gratuito.

**3. PHP:**

* + Lenguaje de programación del lado del servidor.
  + Versión: PHP 7.4.
  + Precio: Gratuito.

**4. MQTT:**

* + Protocolo de mensajería ligero para la transmisión de datos en tiempo real.
  + Precio: Gratuito.
  + Implementación: Se puede implementar utilizando bibliotecas como Mosquitto, HiveMQ, o implementaciones propias en lenguajes como Python, JavaScript, entre otros.

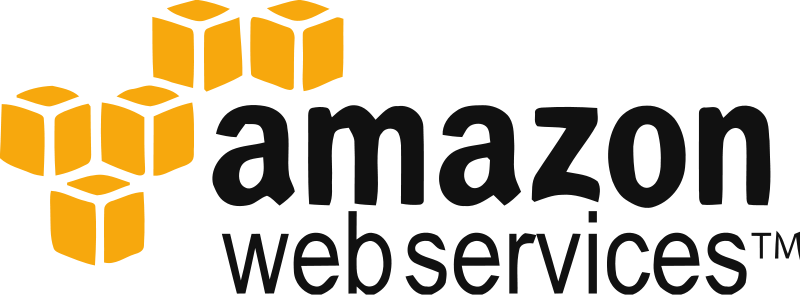


**5. Sistema Operativo Ubuntu 22.04:**

* + Distribución de Linux basada en Debian.
  + Versión: 22.04 (nombre en clave: TBD).
  + Características: Actualizaciones de seguridad regulares, soporte a largo plazo (LTS), entorno de escritorio GNOME, gran comunidad de usuarios y desarrolladores.
  + Precio: Gratuito.

## Servicios:

**1. AWS:**

* + Plataforma de servicios en la nube ofrecida por Amazon.
  + Precio: Variable dependiendo de los servicios utilizados y su escala de uso.

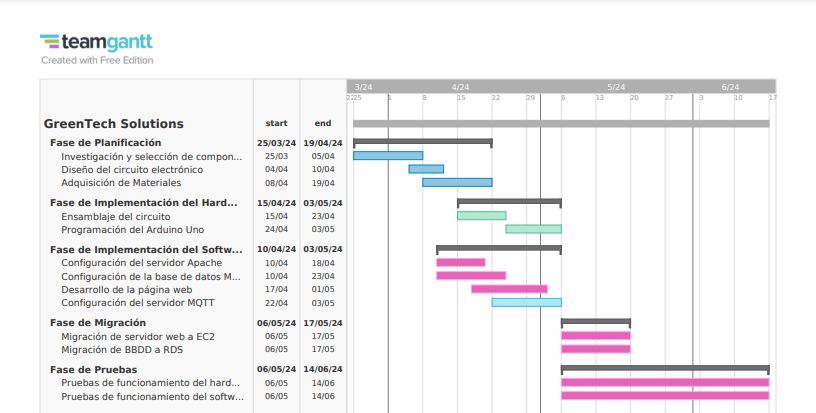
# PRESUPUESTO DETALLADO

Se recogerá el coste de la implantación de la solución propuesta en el proyecto en el contexto indicado.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Componente | URL | Precio  (Sin IVA) | IVA (21%) | Precio (IVA) |
| Arduino UNO R3 | [Link](https://www.amazon.es/Arduino-UNO-A000066-microcontrolador-ATmega328/dp/B008GRTSV6/ref=sr_1_5?crid=3QBQVS22GUC7A&dib=eyJ2IjoiMSJ9.N8nF-4nrSGrUBX0eJZ8DQ3faJEQ6oU7jDfwMD55RC6PHY2xj3N9JePg85PFtW5v-nG1TLbyHvS9oNR5mPbB6sJNW75nQi8cvNy4kP0iT2t9zmy-F27ZnrA6bcqa8AZZNar1W1C-38gvaebG-VIpUeLuLxk2SnyFVRtfiXKLinuQd5iS-ovK5JxbNKNFDLE2FR01vFOMqDdFPUTTZQ6u4iDywcYGhDjz4-p_AGgxXiMcNd0uKmJsMRmqdGunroAgjvu3sF0CX8juMP2dBEHezR08bpnT6785feExFtGKqQeU.Xk6fvyzP2I-TnIgTqIMWK7zQALHMukrGGGX1qA2FSmo&dib_tag=se&keywords=arduino+uno+r3&qid=1713425949&sprefix=arduino+%2Caps%2C102&sr=8-5) | 18,16 € | 4,83 € | 22,99 € |
| Placa de pruebas | [Link](https://www.amazon.es/AZDelivery-Breadboard-barras-conductoras-Arduino/dp/B07KKJSFM1/ref=sr_1_5?crid=2F0BISXJE3DXY&dib=eyJ2IjoiMSJ9.qBPp7lHWWelJcqFJQYHaU0QuRzF7SqCy4PDW4Vus6H3i6FWCMCrJcUbWfocvBWiSrcpqZsns_aj-tGxVm46qMedXsErejA0wjcfdOf1kCLyJ2JrjAkkNHAGPXJnBufM0rvlByYZu-z--8kQSHVK2sqW7Pl0UG4Y7npkQe6TopjizWjn-UbA2-4b2jBGyobo__nowEZfJylMCyXCy5eWudUgpD3jMlgkeemq3v0SgJNp5GFHzYSsZk8ThjJNGFy9rRxNO2spD0ZbTRmkybW_gMfZychMC0zJ6p-fOY6aJTSI.JJS4vaJrBGw3FZS3Cp4WbMK3sYPlUuD8YwZCV7fqLIg&dib_tag=se&keywords=placa+de+pruebas+arduino&qid=1713428538&sprefix=placa+de+pruebas+ar%2Caps%2C103&sr=8-5) | 4,34 € | 1,15 € | 5,49 € |
| Módulo ESP8266-01 | [Link](https://www.amazon.es/AZDelivery-ESP8266-01S-Adaptador-Breadboard-Compatible/dp/B072R6DPK7/ref=sr_1_3?__mk_es_ES=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=1TAMWWNELPA1J&dib=eyJ2IjoiMSJ9.cc3weZljGH3eErmWCpTXT6inbUpVGXjTMvHCPQDanlVyv-lfEfQKtVHB-mWQ3027PwhbugX3tNSZ075GH8jqVoU_kHCSMI0-ubGB-67zyf-iE8JAV_sJnEWS53tfkLHbGqg8VJywmuCpArCUbKxUB7HRdH1OO6a_6zTFO4-xjCmpCH4LVB4kZO-J4egCkphwzgnmJRbW75CDBNCBvSx6Hr80x8cbCesCii2XyeH_AjSlHA8iKsVZDdDhoRb4yLzAfMpUUNlO_LzLoUuRIdzdl3SIaUh36PBz2dC7IsdONOg.i7E123vzvJIdndpLMyY3_Lm74n9o0NPVtW7b41dVWVg&dib_tag=se&keywords=modulo+esp8266-01&qid=1713428466&sprefix=modulo+esp8266-01%2Caps%2C108&sr=8-3) | 5,13 € | 1,36 € | 6,49 € |
| Módulo de alimentación | [Link](https://www.amazon.es/AZDelivery-MB102-Breadboard-Adaptador-alimentaci%C3%B3n/dp/B06X962SPW/ref=sr_1_5?__mk_es_ES=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=TSPVR1GDST12&dib=eyJ2IjoiMSJ9.zJGROkc0oHv54fSXGaAVPepFRb91MUsusVPsob4hadntwTtQgZuOzDh8qfa6Ggd7hxloq8lVl0pj0zP-aW51rBawtr2B8TgxObb38zf8IrMD3C8czRgO1wGbOARpaWxgG2BOvNBWaWAFfe-PS1NXwPi0_2xnHcSpiivieT8VaRvkcWByIiyCc-A-eF1yj9pVOPPxrkgAW-wPKh3xjR8RoGua96pFZqjdvYcJK3KMAoR_Q6SaAiCo8HG-rcUN4MZ7VFTK59QxMb-VxXxGJ5BH1Nl1N4DpBc9FsqaJwfABzG8.c6FgKmMW3m1zpCflNlnzsKlAhCgIWX8bDXYevGGr4Fs&dib_tag=se&keywords=modulo%2Bfuente%2Bde%2Balimentacion%2Barduino&qid=1713428394&sprefix=modulo%2Bfuente%2Bde%2Balimentacion%2Barduino%2Caps%2C97&sr=8-5&th=1) | 3,94 € | 1,05 € | 4,99 € |
| Pack relé + bomba de agua + sensor de humedad del suelo | [Link](https://www.amazon.es/RUNCCI-YUN-Sumergible-Cepillo-Tuber%C3%ADa-paraTanque/dp/B0814HXWVV/ref=sr_1_5?crid=2MUPU3MCNTPQT&dib=eyJ2IjoiMSJ9.BvDnZOOEMe1jbq75fCUEGvu5WhCMollyOuhoSvwHUH2dMISFgI5-Tvh-N20KiBdzR6dcPa12N2VYjiuGCN0jt66Qc2e1J25ESUUNB4InJIQ-skXD7js1nCVtwmA25LruAR6k4HhvuFCGUoV2t_zh7XEpbr06zvCP6qfXkkhCaJuefjSL3wcWu6T2hjXErlVuPX97OvdpH8wM9M_AF3blwQsPwG44LApOT7bFTFkANZKut0WdXX3hcxv_YjzKC7NPsPK-Q5WYUK2xJkGcIhsRITQaHykRvN9oqPdYdsBt500.jVOSEI4JY9ReoKj4jAfbugO4NheDxsUnXwXYSlSqqEE&dib_tag=se&keywords=bomba%2Bde%2Bagua%2Bmini%2Barduino&qid=1713428234&sprefix=mini%2Bbomba%2Bde%2B%2Barduino%2Caps%2C92&sr=8-5&th=1) | 7,89 € | 2,10 € | 9,99 € |
| Sensor DHT11 | [Link](https://www.amazon.es/AZDelivery-Temperatura-Compatible-Arduino-Incluido/dp/B089W8DB5P/ref=sr_1_1_sspa?dib=eyJ2IjoiMSJ9.J8_7o_7nkjidzjJwsOQSVGJTk_DI7naVgLCPtovUVJ7ObflO0e2acy4CTzlM1qZjY0EkKvQ3zgaUXruE6kQBgGJ88zYS0NGs-7J0HgddAVIUDNs8NCl-rW76F0seUGP0QFqYYo8JFtAgv5lILF93f1ZU2sPjxq6MfthSiuoTU9ndeVl0DgVSUwDLCXc5q7lzvQo6QVhoXGwiRi3qs5-6Yd-BooWL4U1QcterRJG73n3VAkXL0sxmcViU5cHgBCCbTmR61kVS9GCb1kN1u747vpsdhJK65mYaTMc6M_XHMho.GTL45kmFU2Jy-XPbw0l0qWKS8N4lllHWz6roPgCiqD4&dib_tag=se&keywords=dht11&qid=1713425766&sr=8-1-spons&sp_csd=d2lkZ2V0TmFtZT1zcF9hdGY&th=1) | 4,73 € | 1,26 € | 5,99 € |
| Total (IVA incluido): | **55,94 €** | | | |

# PLANIFICACIÓN DEL PROYECTO

En este apartado se aporta un diagrama de Gantt con la distribución temporal de todas las tareas desempeñadas para llevar a cabo este proyecto. Se incluirán elementos como la búsqueda y selección del equipamiento, periodos de pruebas.



# DESARROLLO TÉCNICO DEL PROYECTO

## En este apartado se expondrán todos los pasos, procedimientos y enfoques técnicos que se emplearán en el desarrollo del proyecto. Se detallarán las tecnologías, lenguajes de programación y herramientas que se utilizarán. Se describirá el diseño detallado de las funcionalidades del sistema, utilizando imágenes para aclarar las interacciones entre los componentes. Además, se expondrá el enfoque y las prácticas de desarrollo de software que se emplearán, junto con los procesos y herramientas para la integración continua y el despliegue continuo del software. En resumen, este apartado proporcionará una guía técnica integral para la ejecución exitosa del proyecto.

## Configuración del Hardware

### Ensamblaje y Cableado de los Componentes

### Configuración de la Fuente de Alimentación

En este proyecto, se está modificando una fuente de alimentación ATX para adaptarla a las necesidades específicas del sistema, ya que el Arduino y otros sensores no pueden alimentarse adecuadamente con los voltajes estándar de 12V de una fuente ATX. En lugar de adquirir una fuente de alimentación independiente que opere en los voltajes adecuados (5V y 3.3V) y proporcione suficiente corriente, se ha optado por utilizar una fuente ATX existente y reconfigurarla.

Modificar una fuente de alimentación ATX para utilizar solo los voltajes de 5V y 3.3V mientras se mantiene el funcionamiento de la fuente es un proceso relativamente sencillo, pero requiere precaución y atención a los detalles para evitar daños tanto en la fuente como en los dispositivos que se alimentarán con ella. Aquí tienes una explicación detallada del proceso:

**Materiales Necesarios:**

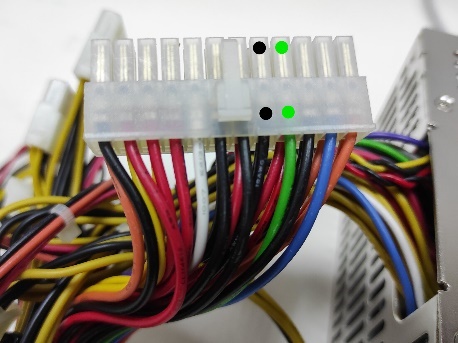
* Fuente de alimentación ATX.
* Conectores y cables adicionales si es necesario.
* Soldador y estaño.
* Cable pelacables.
* Cinta aislante o termorretráctil.
* Multímetro para comprobaciones.

**Paso 1: Preparación**

1. Desconecta la fuente de alimentación de cualquier toma de corriente y asegúrate de que esté apagada.
2. Abre la carcasa de la fuente de alimentación con un destornillador adecuado.

**Paso 2: Identificación de Cables**

1. Localiza los cables de salida de 5V (cables rojos) y 3.3V (cables naranjas) en el conector principal de la placa base.
2. Identifica el cable verde (PS\_ON) y los cables negros (GND) en el mismo conector.



**Paso 3: Desconexión de Cables**

1. Corta o desuelda los cables de salida de 12V del conector principal de la placa base, dejando solo los cables de 5V y 3.3V.
2. Si es necesario, corta y aísla los cables de 12V en la fuente de alimentación para evitar cortocircuitos.

**Paso 4: Empalme del Cable Verde (PS\_ON)**

1. Corta un cable negro de longitud suficiente para alcanzar el conector del cable verde (PS\_ON).
2. Realiza un empalme (puente) entre el cable verde (PS\_ON) y el cable negro utilizando soldadura y estaño.
3. Asegura el empalme con cinta aislante o termorretráctil para evitar cortocircuitos.

**Paso 5: Prueba y Comprobación**

1. Vuelve a montar la carcasa de la fuente de alimentación.
2. Conecta la fuente de alimentación a una toma de corriente y enciéndela.
3. Utiliza un multímetro para comprobar los voltajes de salida de 5V y 3.3V y asegúrate de que estén dentro de los límites especificados.
4. Verifica que la fuente de alimentación funcione correctamente y que los dispositivos alimentados reciban la energía adecuada.

### Desarrollo del Código para Arduino

### Verificación del Funcionamiento del Hardware

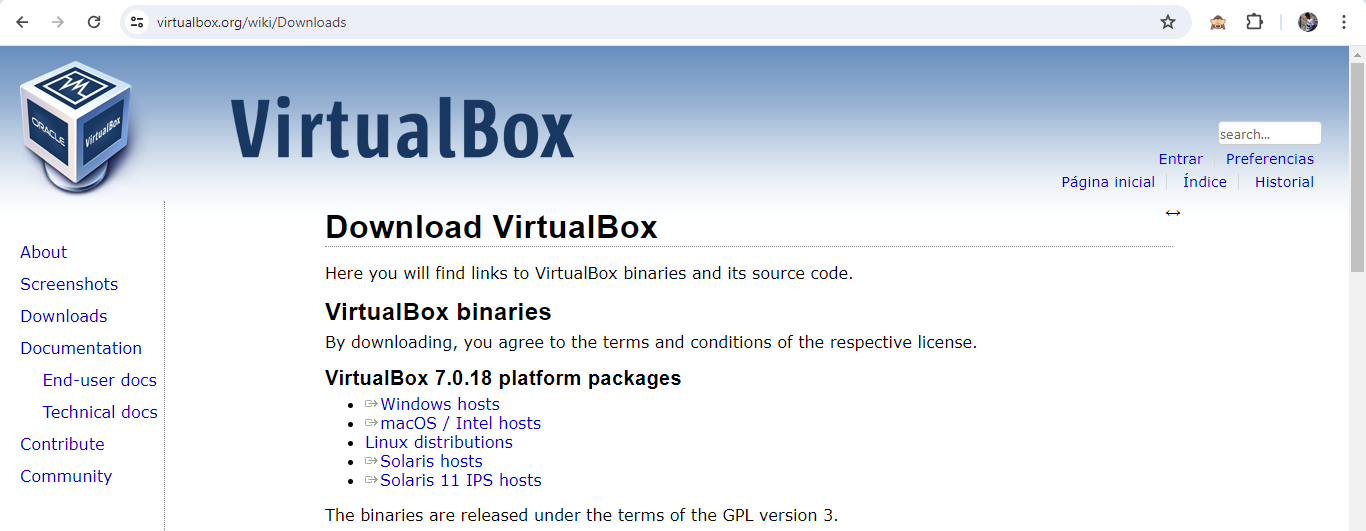
## Implementación en Entorno Local

### Preparación del Entorno de Desarrollo (Instalación de Ubuntu 22.04)

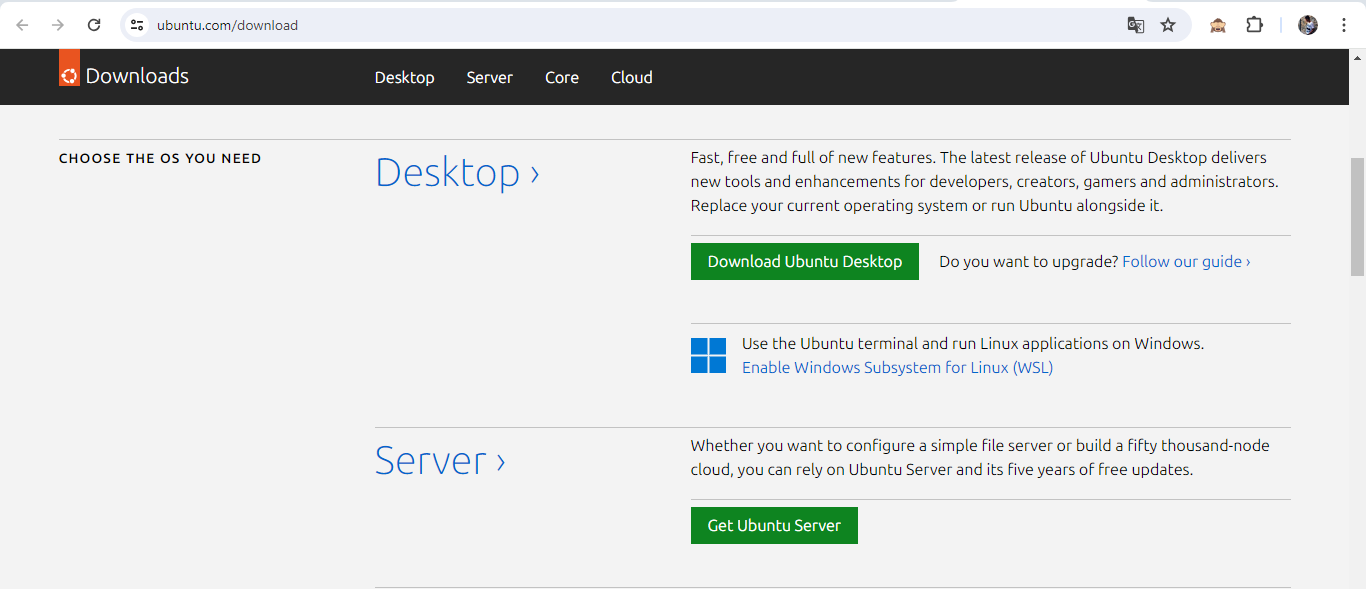
La instalación de Ubuntu se llevará a cabo en un entorno virtual utilizando VirtualBox, lo que nos permitirá realizar pruebas y experimentos sin comprometer un sistema operativo físico. Una vez completadas y verificadas las configuraciones en VirtualBox, el siguiente paso será trasladar todo el entorno a AWS para su despliegue en producción. Durante la instalación en VirtualBox, configuraremos los aspectos básicos del sistema operativo, como el particionamiento del disco, la configuración de usuarios y contraseñas, así como la instalación de paquetes de software necesarios.

**Paso 1: Configurar VirtualBox**

**Descarga e instala VirtualBox**: Ve al sitio web de VirtualBox (<https://www.virtualbox.org/>) y descarga la versión adecuada para tu sistema operativo. Sigue las instrucciones de instalación proporcionadas por el instalador.



**Descarga la imagen ISO de Ubuntu**: Ve al sitio web oficial de Ubuntu (<https://ubuntu.com/download>) y descarga la imagen ISO de Ubuntu 22.04 LTS.

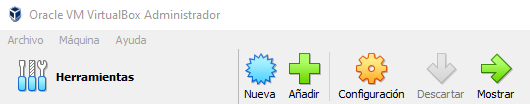


**Paso 2: Crear una nueva máquina virtual**

**Abre VirtualBox**: Ejecuta VirtualBox desde el menú de inicio o desde la ubicación donde lo hayas instalado.

**Crea una nueva máquina virtual**:

Haz clic en el botón "Nuevo" en la barra de herramientas.



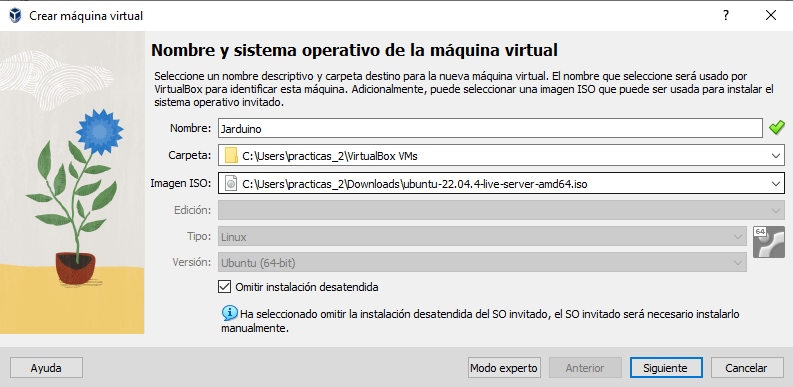
En el asistente de creación de máquinas virtuales, ingresa un nombre para la máquina virtual.

Selecciona el tipo de sistema operativo como "Linux".

Selecciona la versión como "Ubuntu (64-bit)" si has descargado la versión de 64 bits, o la versión de 32 bits según corresponda.

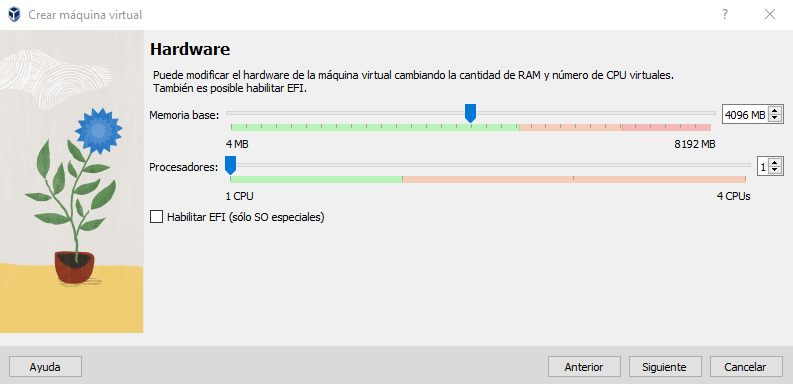
Puedes omitir estos pasos si seleccionamos la ISO ahora.

Haz clic en "Siguiente".



**Paso 3: Configurar la máquina virtual**

**Asigna memoria RAM**: En el siguiente paso del asistente, asigna la cantidad de memoria RAM que deseas para la máquina virtual. Se recomiendan al menos 2 GB para Ubuntu.

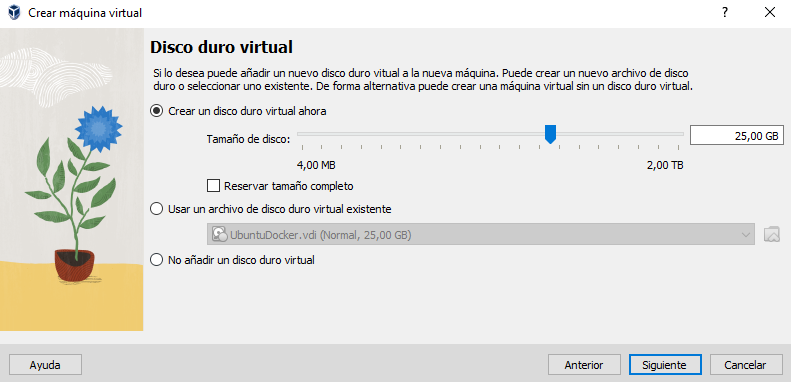


**Crea un disco duro virtual**:

Selecciona "Crear un disco duro virtual ahora".

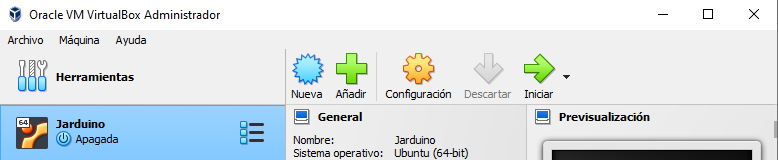
Especifica el tamaño del disco duro virtual. Se recomiendan al menos 20 GB para una instalación básica de Ubuntu.

Haz clic en "Crear" para finalizar la configuración del disco duro.



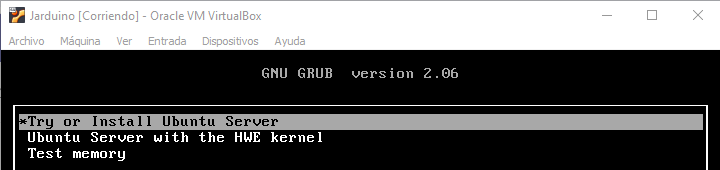
**Paso 4: Instalar Ubuntu en la máquina virtual**

**Inicia la máquina virtual**: Selecciona la máquina virtual y haz clic en "Iniciar" en la barra de herramientas.

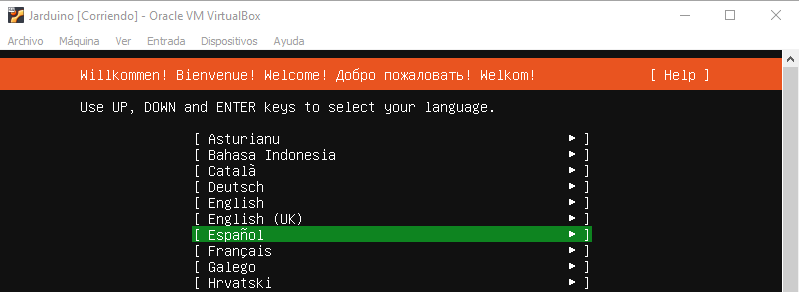


**Arranca desde la imagen ISO**: La máquina virtual debería iniciarse desde la imagen ISO de Ubuntu que configuraste. Si no lo hace, asegúrate de que la imagen esté seleccionada en la configuración de la máquina virtual.

**Selecciona "Instalar Ubuntu"** en el menú de arranque.

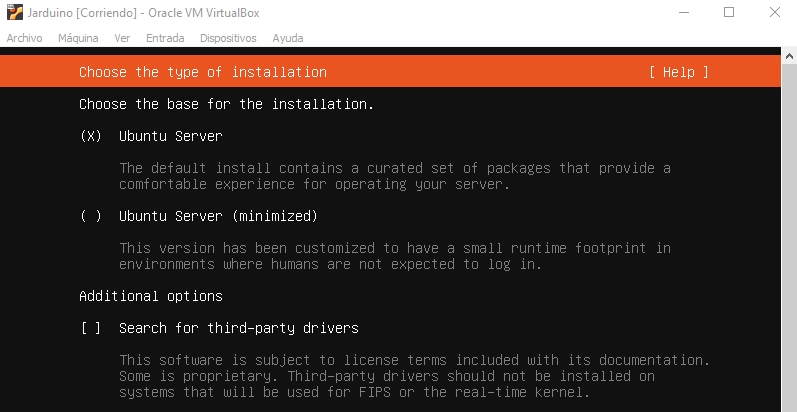


**Sigue el instalador de Ubuntu**: Sigue los pasos del instalador de Ubuntu para seleccionar el idioma, la ubicación, el tipo de teclado y otros ajustes.

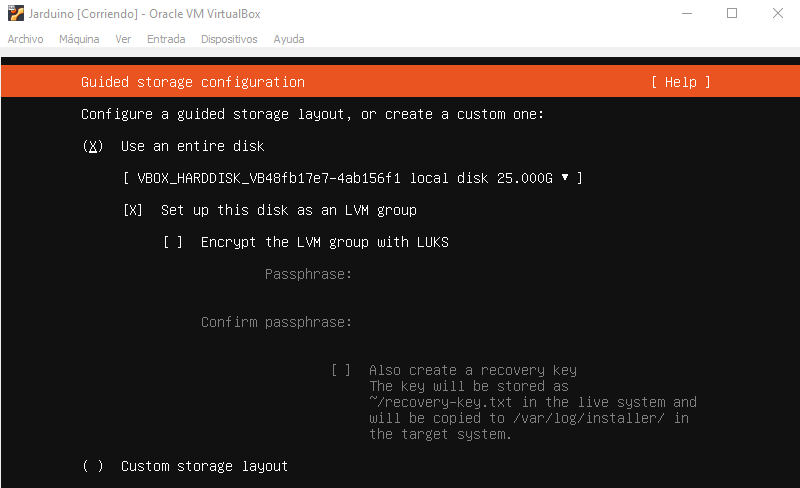


**Configura la instalación**:

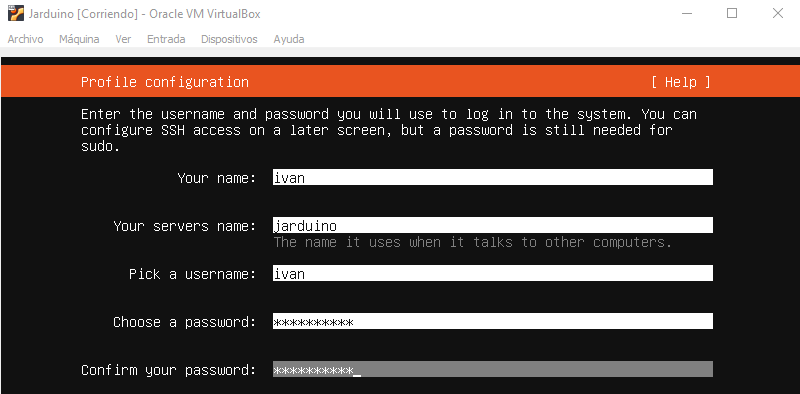
Selecciona la opción de instalación normal y elige si deseas instalar actualizaciones y software de terceros durante la instalación.



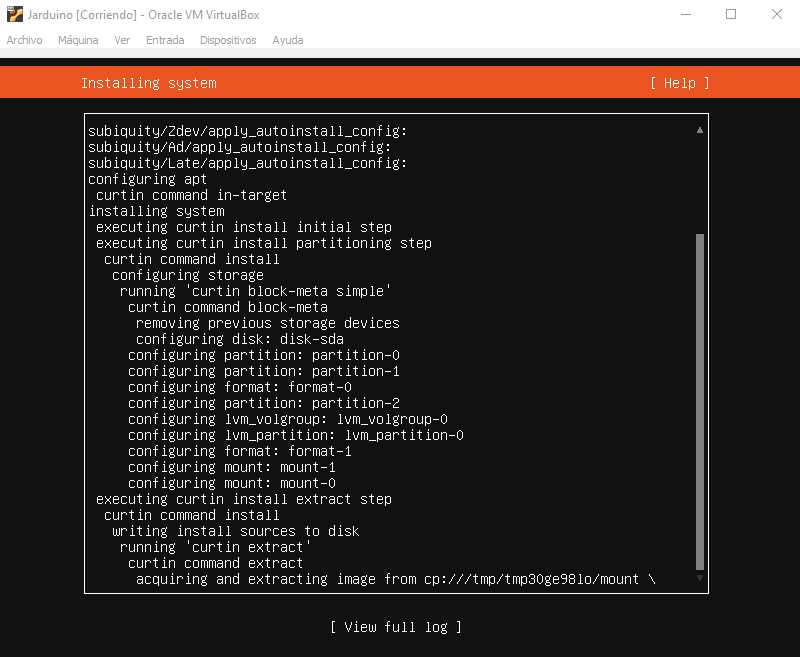
Elige el método de instalación del disco duro (puedes dejar la opción predeterminada para usar todo el disco si es para propósitos de prueba).



Crea una cuenta de usuario y establece una contraseña.



**Finaliza la instalación**: Una vez que el instalador haya completado la instalación, reinicia la máquina virtual y selecciona "Iniciar Ubuntu" desde el menú de arranque.



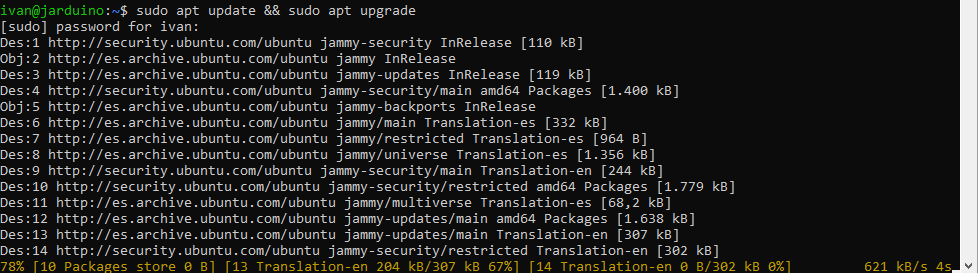
### Configuración del Servidor Web (Instalación de Apache, PHP y MySQL)

En el proceso de configuración de un servidor web en Ubuntu Server 22.04, se requiere la instalación de componentes clave que permitirán alojar y ejecutar aplicaciones web de manera efectiva. Para ello, se instalarán y configurarán Apache, un servidor web de código abierto ampliamente utilizado; PHP, un lenguaje de programación popular para el desarrollo web dinámico; y MySQL, un sistema de gestión de bases de datos relacional. Estos componentes proporcionarán una base sólida para el desarrollo y la ejecución de aplicaciones web en el entorno Ubuntu Server, permitiendo el almacenamiento, procesamiento y entrega de contenido web de manera eficiente y segura.

**Actualizar el sistema**

Antes de comenzar cualquier instalación de software, es crucial asegurarse de que el sistema esté actualizado para obtener las últimas correcciones de seguridad y actualizaciones. Puedes hacerlo ejecutando los siguientes comandos en la terminal:

sudo apt update && sudo apt upgrade

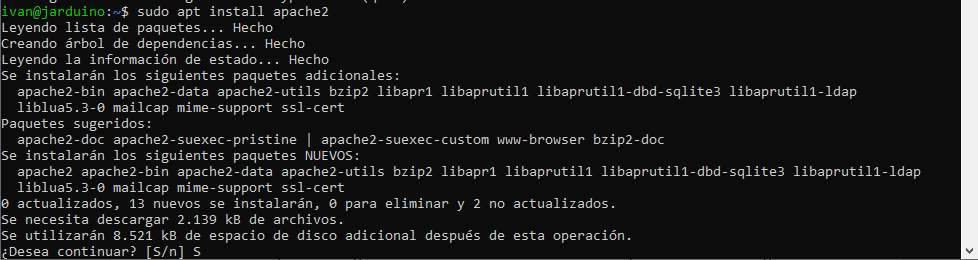


Esto actualizará los repositorios de software y aplicará las actualizaciones disponibles en tu sistema.

**Instalar Apache**

Apache es un servidor web de código abierto ampliamente utilizado en Internet. Para instalar Apache en Ubuntu, simplemente ejecuta el siguiente comando:

sudo apt install apache2

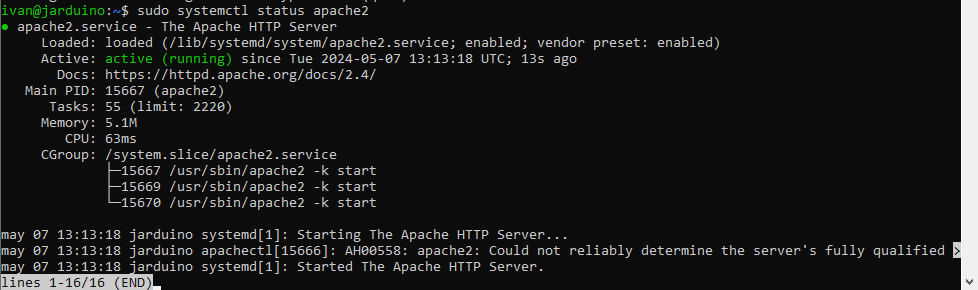


Después de ingresar tu contraseña de administrador, el sistema descargará e instalará Apache junto con todas sus dependencias. Una vez completada la instalación, Apache debería estar automáticamente configurado para iniciarse en el arranque del sistema.

**Verificar el estado de Apache**

Después de la instalación, puedes verificar el estado de Apache para asegurarte de que esté en funcionamiento correctamente. Ejecuta el siguiente comando:

sudo systemctl status apache2

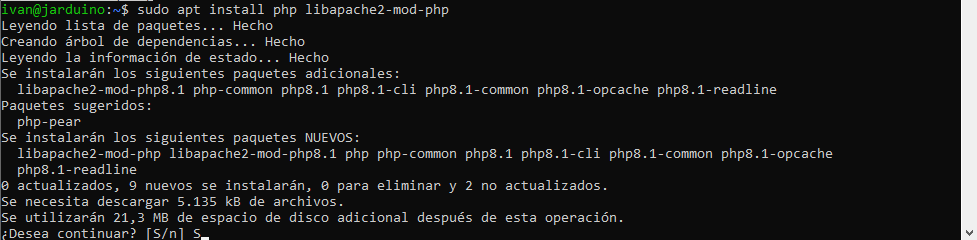


Si Apache se está ejecutando correctamente, verás un mensaje indicando que el servicio está activo y en funcionamiento.

**Instalar PHP**

PHP es un lenguaje de programación ampliamente utilizado para desarrollar aplicaciones web dinámicas. Puedes instalar PHP en Ubuntu junto con el módulo Apache necesario utilizando el siguiente comando:

sudo apt install php libapache2-mod-php

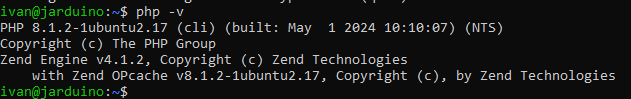


Una vez completada la instalación, PHP estará integrado con Apache y listo para su uso.

**Verificar la instalación de PHP**

Después de instalar PHP, es importante verificar que esté configurado correctamente. Puedes hacerlo ejecutando el siguiente comando:

php -v

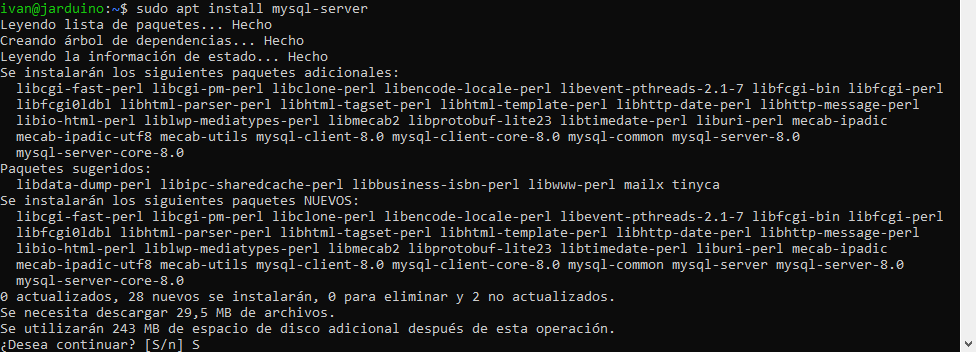


Esto mostrará la versión de PHP instalada en tu sistema y confirmará que PHP está funcionando correctamente.

**Instalar MySQL**

MySQL es un sistema de gestión de bases de datos relacional utilizado para almacenar y administrar datos. Puedes instalar MySQL en Ubuntu utilizando el siguiente comando:

sudo apt install mysql-server

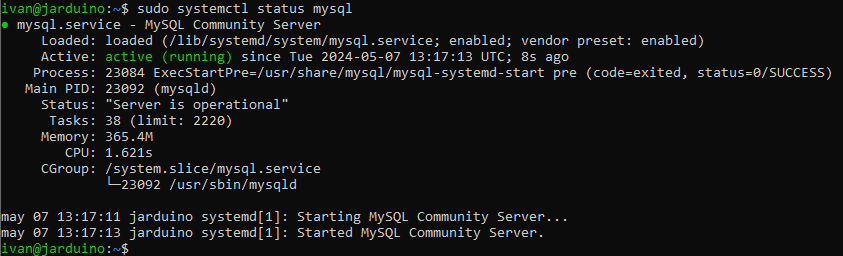


Durante la instalación, se te pedirá que configures una contraseña para el usuario root de MySQL. Asegúrate de elegir una contraseña segura y memorízala o guárdala en un lugar seguro.

**Verificar la instalación de MySQL**

Después de instalar MySQL, puedes verificar su estado para asegurarte de que esté en funcionamiento correctamente. Ejecuta el siguiente comando:

sudo systemctl status mysql

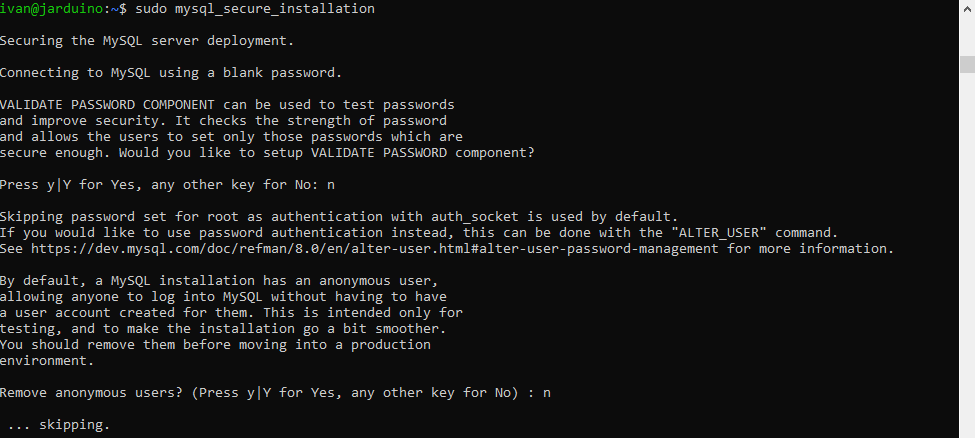


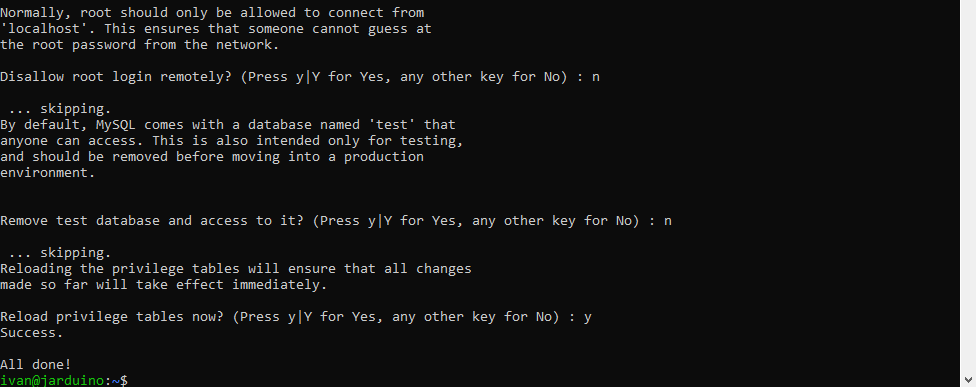
Si MySQL se está ejecutando correctamente, verás un mensaje indicando que el servicio está activo y en funcionamiento.

**Configurar MySQL**

Una vez instalado MySQL, es recomendable ejecutar el script de seguridad incluido para asegurar la instalación y configuración básica de MySQL. Puedes hacerlo ejecutando el siguiente comando:

sudo mysql\_secure\_installation





Este script te guiará a través de varios pasos para mejorar la seguridad de tu instalación de MySQL, como establecer una contraseña segura para el usuario root, eliminar usuarios anónimos, etc.

**Reiniciar Apache**

Después de instalar PHP y MySQL, es recomendable reiniciar el servidor Apache para que los cambios surtan efecto. Puedes hacerlo ejecutando el siguiente comando:

sudo systemctl restart apache2



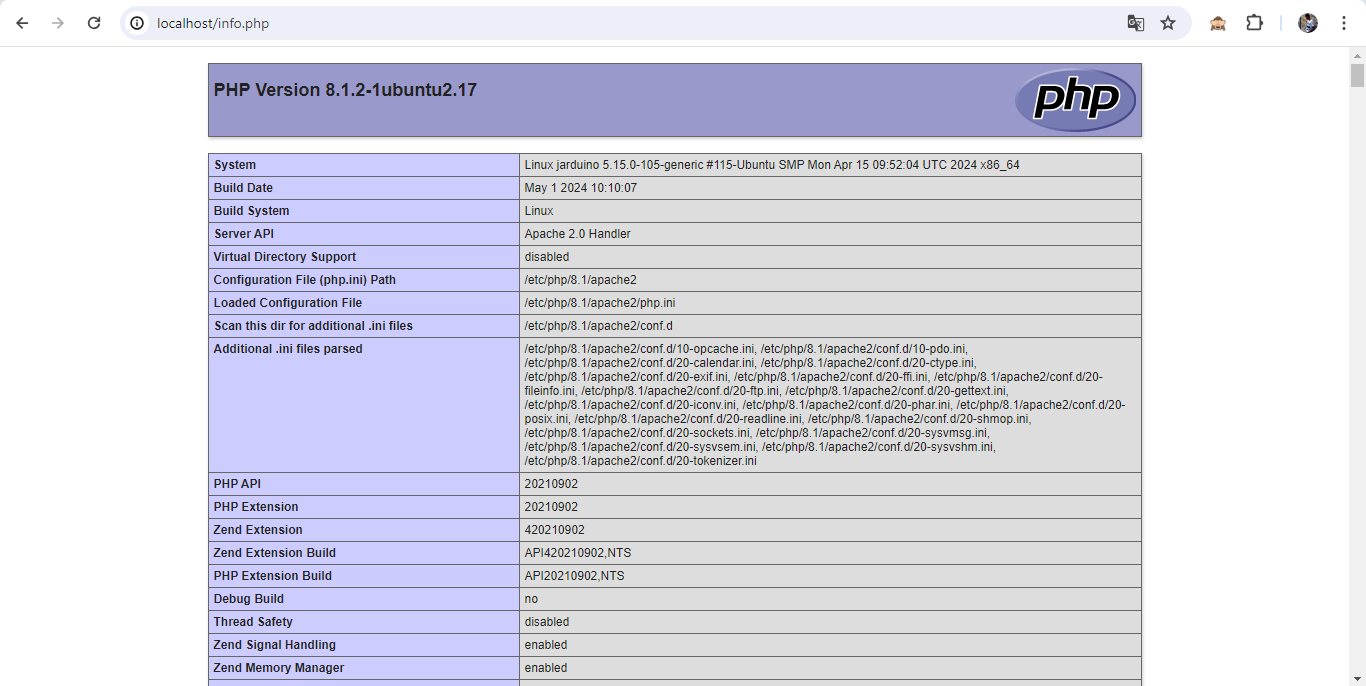
**Verificar la instalación completa**

Para asegurarte de que Apache, PHP y MySQL estén instalados y funcionando correctamente, puedes crear un archivo PHP de prueba y acceder a él desde tu navegador web.

echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/info.php



Luego, abre tu navegador web y accede a **http://tu\_dirección\_ip/info.php**. Deberías ver una página con información detallada sobre la configuración de PHP.



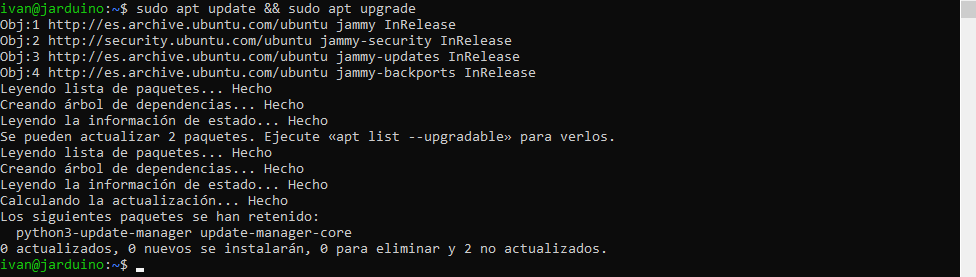
### Integración de MQTT con WebSocket (Instalación y Configuración de Mosquitto)

Vamos a instalar y configurar Mosquitto, un servidor MQTT de código abierto, junto con el soporte de WebSockets en un servidor Ubuntu 22.04. MQTT es un protocolo de mensajería ligero diseñado para dispositivos con recursos limitados y aplicaciones de Internet de las cosas (IoT). La integración de WebSockets permite a los clientes establecer conexiones bidireccionales en tiempo real con el servidor MQTT a través de conexiones web estándar. Este proceso es esencial para habilitar la comunicación entre dispositivos IoT y aplicaciones web de forma eficiente y segura.

**Agregar el repositorio de Mosquitto**

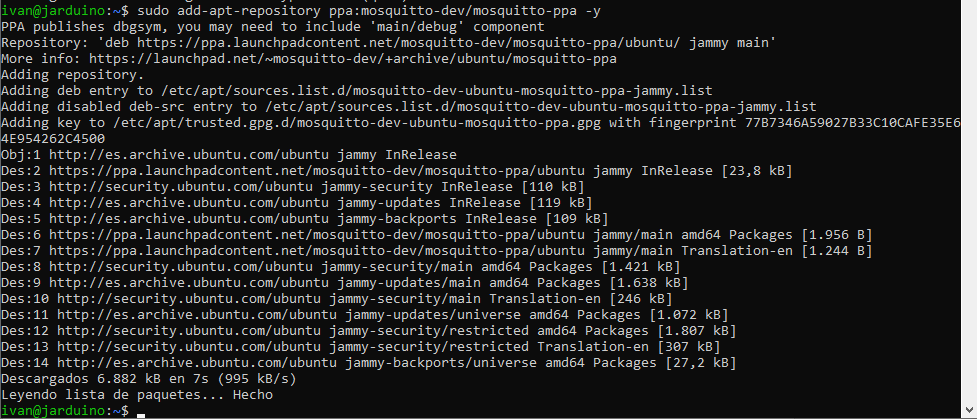
**Actualizar el sistema**: Antes de comenzar, es importante asegurarse de que el sistema esté actualizado para garantizar que todas las dependencias estén al día y se puedan instalar correctamente. Ejecuta los siguientes comandos:

sudo apt update && sudo apt upgrade



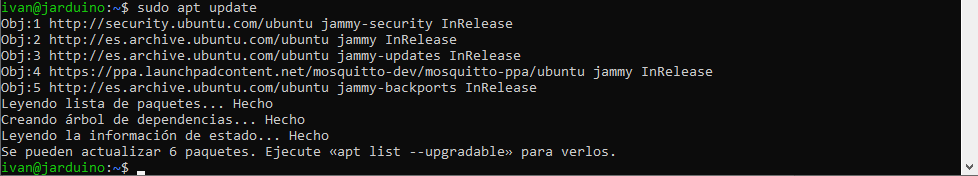
**Agregar el repositorio de Mosquitto**: Dado que Mosquitto no está incluido en los repositorios predeterminados de Ubuntu 22.04, necesitamos agregar el repositorio oficial de Mosquitto. Esto se hace para asegurar que obtengamos la última versión estable y compatible con nuestra versión de Ubuntu. Primero, importamos la clave GPG del repositorio para verificar la autenticidad de los paquetes y luego agregamos el repositorio a nuestra lista de fuentes de software. Ejecuta los siguientes comandos:

sudo add-apt-repository ppa:mosquitto-dev/mosquitto-ppa -y



**Actualizar los repositorios**: Una vez agregado el repositorio de Mosquitto, necesitamos actualizar la lista de paquetes para incluir los paquetes de Mosquitto. Esto se hace ejecutando el siguiente comando:

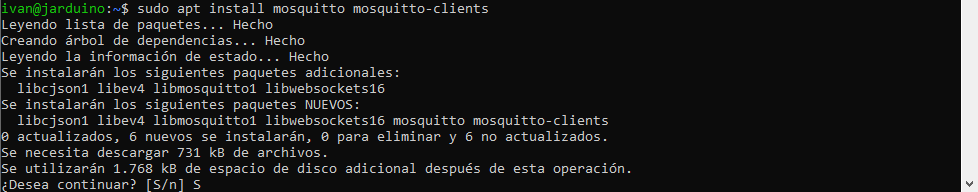
sudo apt update



**Instalar Mosquitto**

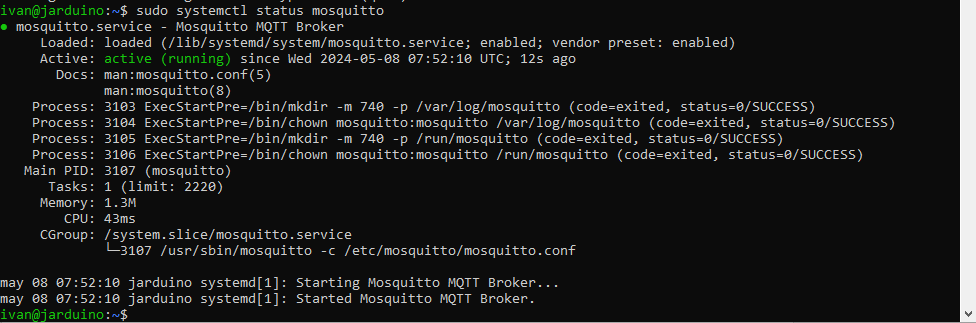
**Instalar Mosquitto**: Ahora que hemos agregado el repositorio de Mosquitto, podemos instalar Mosquitto y sus herramientas de cliente. Esto se logra ejecutando el siguiente comando:

sudo apt install mosquitto mosquitto-clients



**Verificar la instalación**: Una vez completada la instalación, podemos verificar si Mosquitto se está ejecutando correctamente ejecutando el siguiente comando:

sudo systemctl status mosquitto



**Configurar Mosquitto**

**Editar el archivo de configuración de Mosquitto**: La configuración predeterminada de Mosquitto debería ser suficiente para la mayoría de los casos de uso, pero si necesitas ajustar la configuración, puedes hacerlo editando el archivo de configuración ubicado en **/etc/mosquitto/mosquitto.conf**. Puedes usar cualquier editor de texto, como nano, para editar el archivo:

sudo nano /etc/mosquitto/mosquitto.conf

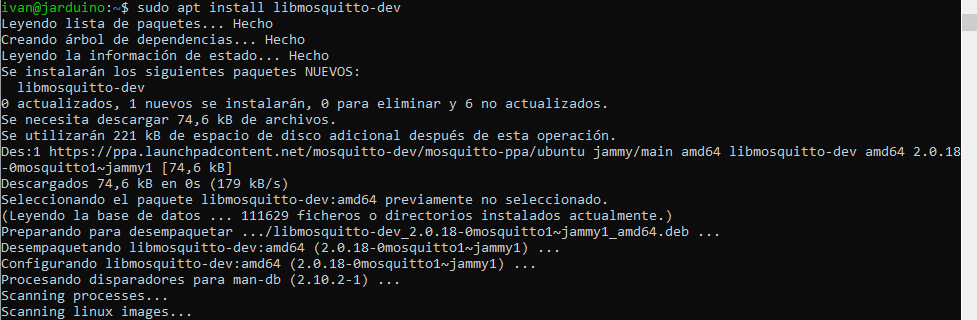
**Reiniciar Mosquitto**: Después de realizar cambios en la configuración, es importante reiniciar el servicio de Mosquitto para aplicar los cambios. Esto se puede hacer ejecutando el siguiente comando:

sudo systemctl restart mosquitto

**Instalar y configurar MQTT con WebSockets**

**Instalar el módulo WebSockets para Mosquitto**: Para habilitar el soporte de WebSockets en Mosquitto, necesitamos instalar el módulo correspondiente. Esto se puede hacer instalando el paquete **libmosquitto-dev**:

sudo apt install libmosquitto-dev



**Editar el archivo de configuración de Mosquitto**: Una vez instalado el módulo, necesitamos editar nuevamente el archivo de configuración de Mosquitto para habilitar explícitamente el protocolo WebSocket y especificar los puertos en los que escuchará las conexiones WebSocket. Abre el archivo de configuración con tu editor de texto preferido:

sudo nano /etc/mosquitto/mosquitto.conf

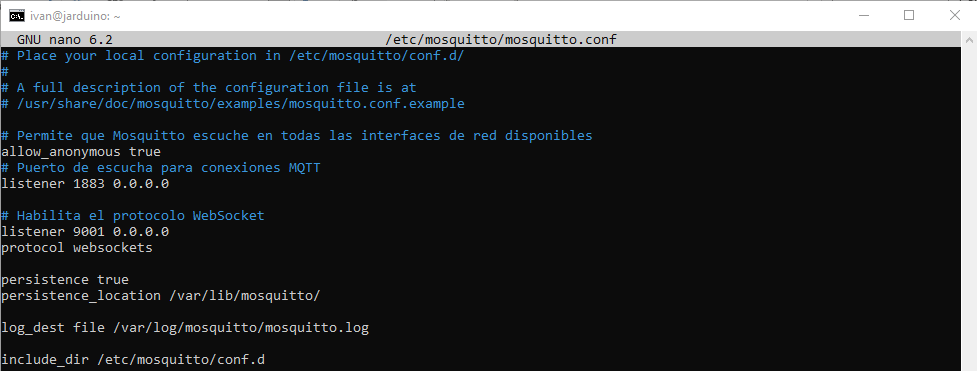
Luego, agrega las siguientes líneas al archivo:

allow\_anonymous true

listener 1883 0.0.0.0

listener 9001 0.0.0.0

protocol websockets



**Reiniciar Mosquitto**: Para que los cambios surtan efecto, es necesario reiniciar el servicio de Mosquitto:

sudo systemctl restart mosquitto

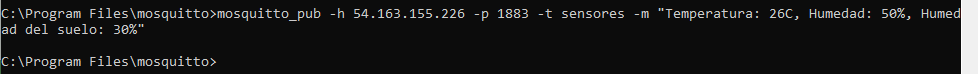


**Verificar la configuración**

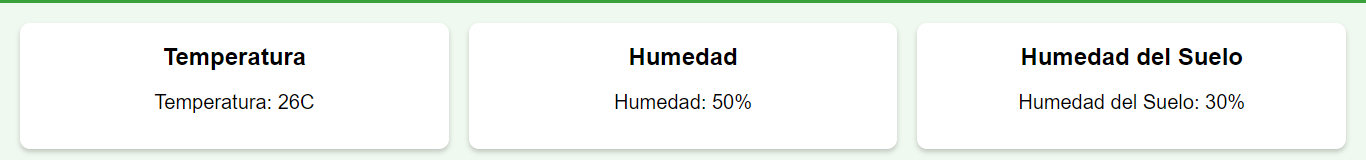
**Verificar los puertos abiertos**: Asegúrate de que los puertos 1883 y 9001 estén abiertos en el cortafuegos de tu servidor para permitir conexiones entrantes a Mosquitto.

**Probar la conexión**: Utiliza un cliente MQTT que admita WebSockets para conectarte a Mosquitto a través del puerto 9001. Vamos a crear una página web con JavaScript para enviar y recibir mensajes MQTT a través de WebSockets.

Enviamos unos datos



Y ahora comprobamos en la página web como los recibe y los muestra



### Desarrollo de la Página Web y Base de Datos

### Pruebas de Conexión entre Arduino y la Página Web

## Migración a la Nube (AWS)

### Creación de una Infraestructura Virtual Privada (VPC)

Una Virtual Private Cloud (VPC) es un servicio de AWS que te permite crear una red virtual aislada en la nube. Funciona de manera similar a una red tradicional en un centro de datos físico, pero con la flexibilidad y escalabilidad que ofrece la nube. Con una VPC, puedes definir tus propias subredes, configurar las reglas de enrutamiento y controlar la seguridad de la red.

Se trata del primer paso para la creación de cualquier elemento en línea de AWS. A través de la opción VPC y más se desplegará todo lo necesario para establecer una VPC con las especificaciones deseadas.

**Pasos Detallados para Crear una VPC en AWS:**

**Inicia sesión en la Consola de AWS:**

Accede a la consola de AWS con tus credenciales de inicio de sesión en <https://aws.amazon.com/>.

**Navega a la Consola de VPC:**

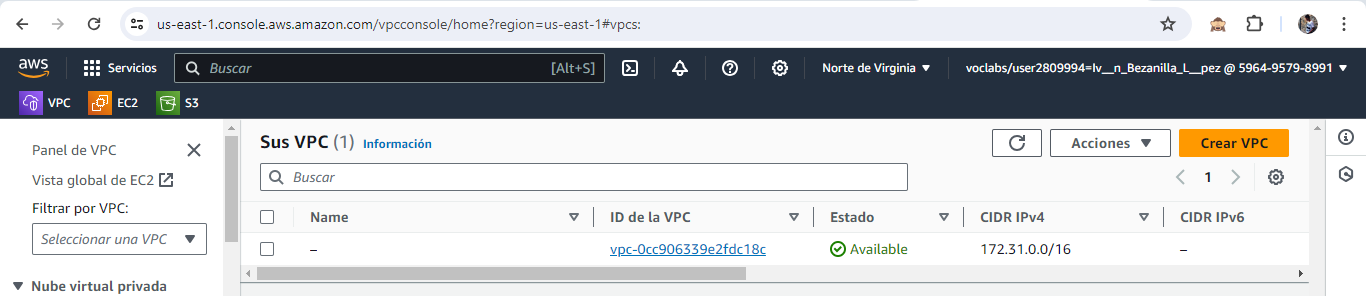
En la barra de búsqueda, escribe "VPC" y selecciona "VPC" en los resultados.

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

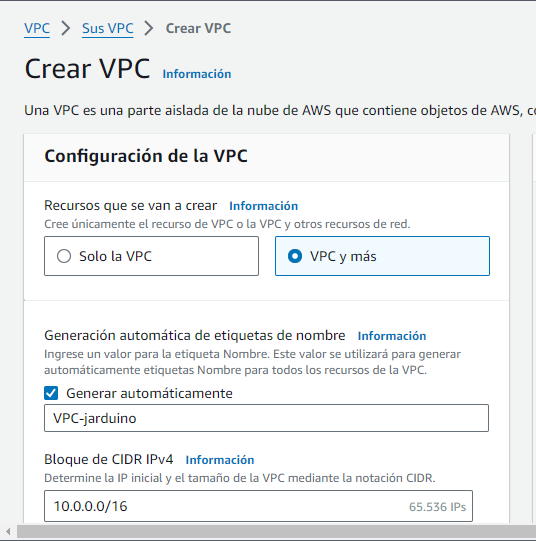
**Inicia el Asistente de Creación de VPC:**

En la página de VPC, haz clic en "Crear VPC".



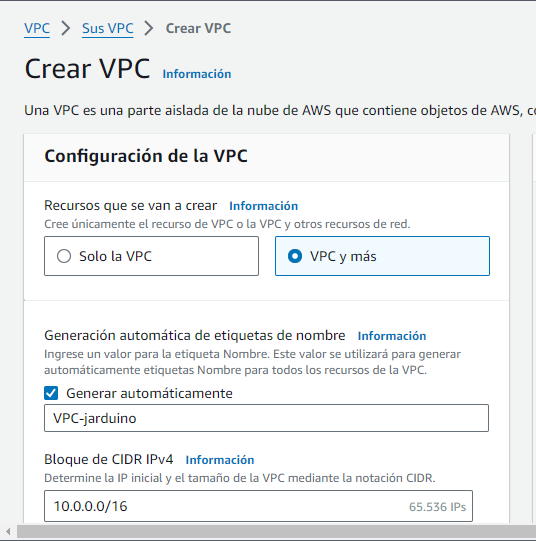
**Selecciona una Configuración de Red:**

Puedes elegir entre varias configuraciones predefinidas (por ejemplo, una VPC con una única subnet pública) o seleccionar "VPC con subredes públicas y privadas".

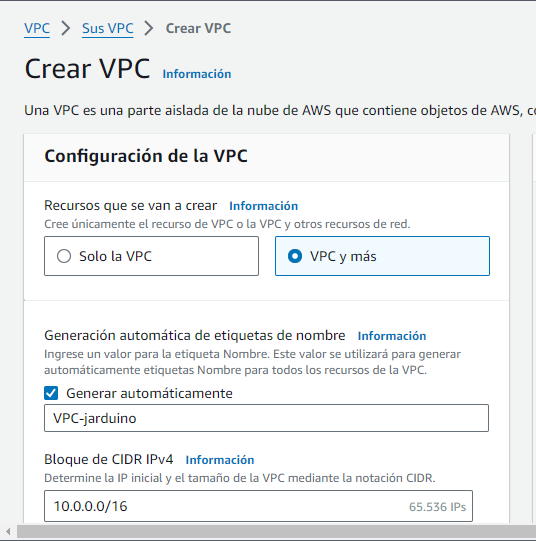


**Configura los Detalles de la VPC:**

Proporciona un nombre para tu VPC.

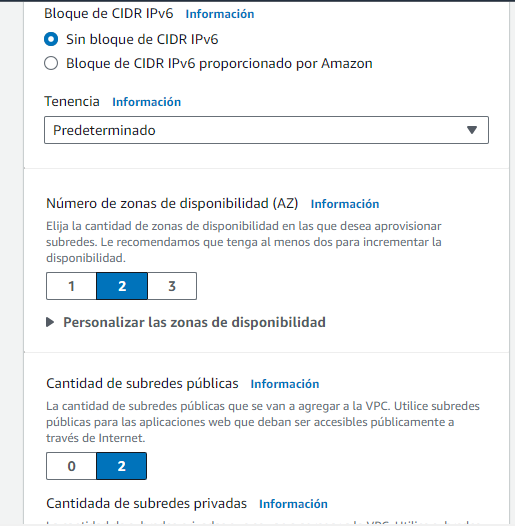


Ingresa el bloque de direcciones IP para la VPC (en formato CIDR, por ejemplo, 10.0.0.0/16).



**Configura las Subredes:**

Define las subredes dentro de la VPC. Puedes especificar subredes públicas y privadas.



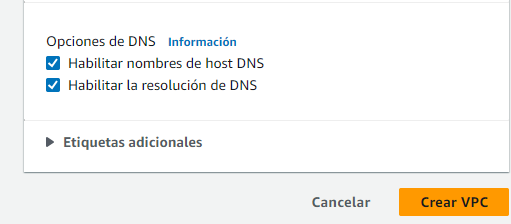


**Revisa la Configuración:**

Revisa todas las configuraciones antes de crear la VPC para asegurarte de que todo esté configurado según tus necesidades.

**Crea la VPC:**

Haz clic en "Crear VPC" para finalizar el proceso.



### Configuración de Grupos de Seguridad

Un grupo de seguridad en AWS es un mecanismo de control de acceso que actúa como un firewall virtual para tus recursos en la nube. Es una parte fundamental de la configuración de la red dentro de una VPC (Virtual Private Cloud) y te permite especificar las reglas de tráfico de red entrante y saliente para tus instancias EC2 (Elastic Compute Cloud), bases de datos RDS (Relational Database Service), y otros servicios de AWS.

Una vez creada la VPC, se dispondrá de la pestaña de grupos de seguridad, donde se pueden crear nuevos grupos y editar las reglas de entrada y salida de información dentro de nuestra red. Todos los EC2 relacionados con el proyecto serán incluidos en el mismo grupo de seguridad. Este grupo de seguridad tendrá permitido cualquier salida de datos, pero tendrá acceso restringido a sus reglas de entrada, teniendo permiso los puertos HTTP 80, SSH 22, MySQL 3306 y HTTPS 443.

**Pasos Detallados para Crear un Grupo de Seguridad en AWS:**

**Navega a la Consola de VPC:**

En la barra de búsqueda, escribe "VPC" y selecciona "VPC" en los resultados.

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

**Ve a la sección "Grupos de seguridad" en la barra lateral izquierda:**

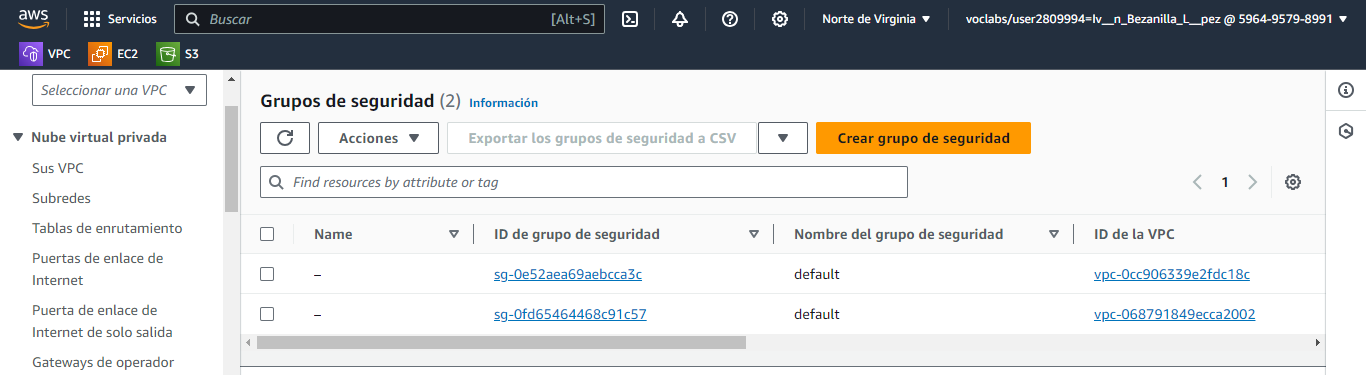
En el panel de navegación izquierdo, busca y selecciona "Grupos de seguridad" bajo la sección "Seguridad".

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

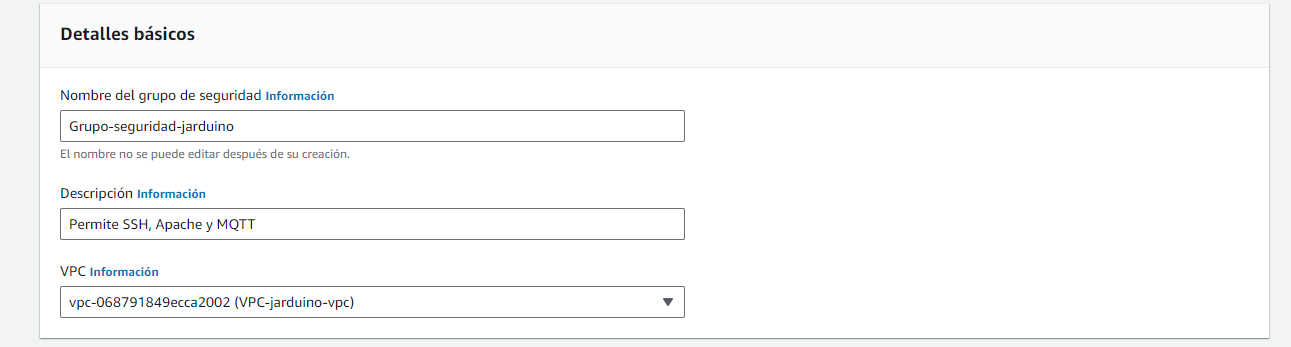
**Haz clic en el botón "Crear grupo de seguridad":**

En la parte superior de la página "Grupos de seguridad", haz clic en el botón "Crear grupo de seguridad".

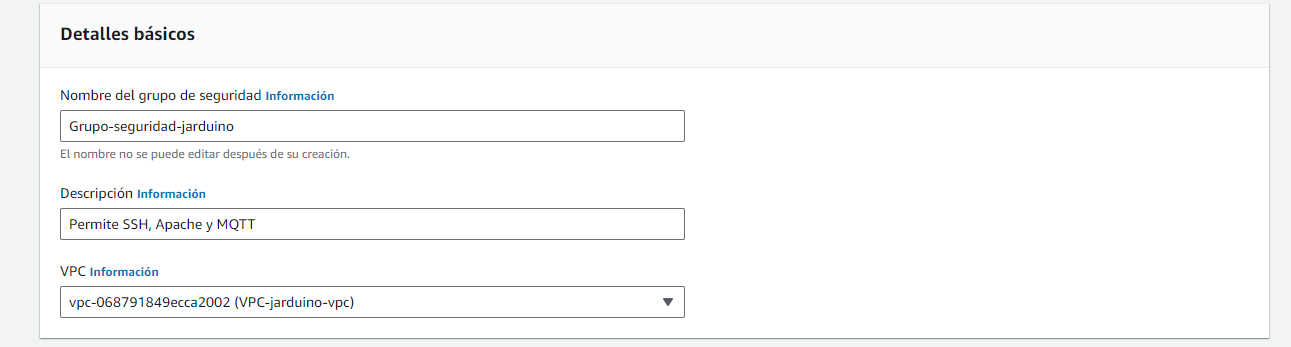


**Configura los detalles básicos:**

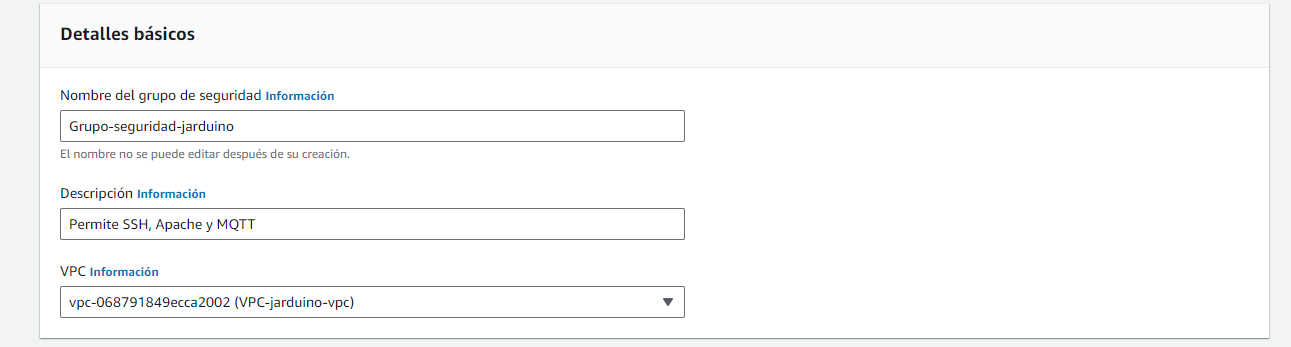
Ingresa un nombre descriptivo para tu grupo de seguridad.



Proporciona una descripción opcional.



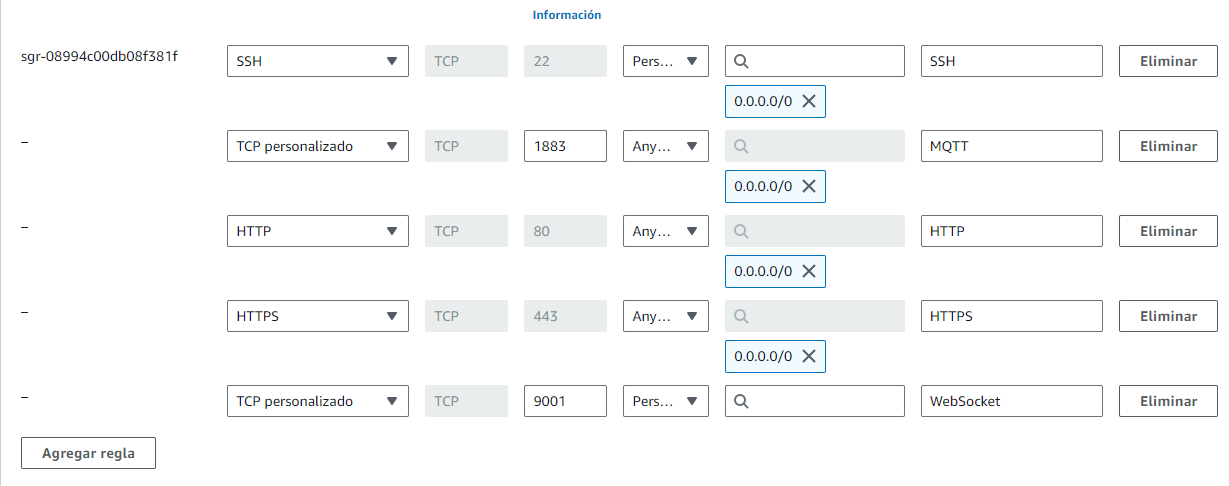
Proporciona la VPC en la cual quieres asignar este grupo de seguridad



**Configura las reglas de entrada:**

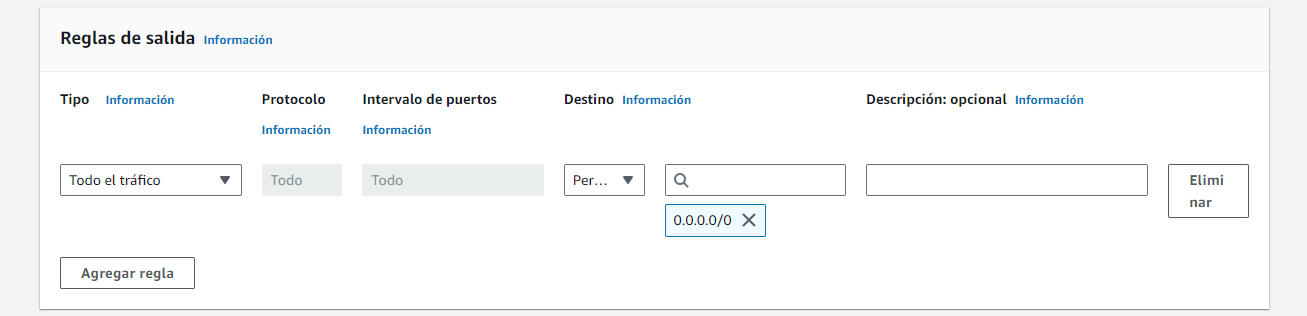
En la sección "Reglas de entrada", especifica las reglas para permitir el tráfico entrante. Por ejemplo, puedes permitir el tráfico SSH (para conexiones SSH), el tráfico HTTP (para conexiones web), el tráfico MQTT (para mosquitto) y el tráfico para el servidor WebSocket.

Define el tipo de tráfico (SSH, HTTP, etc.), el rango de direcciones IP permitido y el puerto.



**Configura las reglas de salida (opcional):**

En la sección "Reglas de salida", puedes especificar reglas para el tráfico que sale de las instancias.

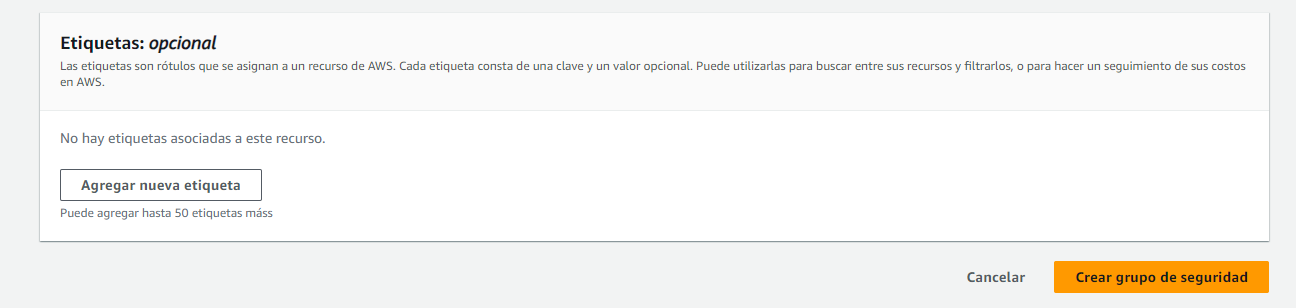


**Revisa la configuración:**

Revisa todas las configuraciones para asegurarte de que sean correctas.

**Haz clic en "Crear grupo de seguridad":**

Una vez que estés satisfecho con la configuración, haz clic en el botón "Crear grupo de seguridad".



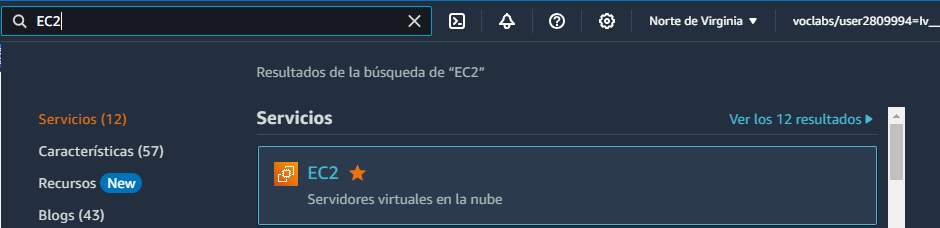
### Implementación de Instancias EC2 y Servicios Asociados

Una instancia EC2 (Elastic Compute Cloud) es un servicio de AWS que te permite lanzar y gestionar servidores virtuales en la nube. Estas instancias pueden ejecutar una amplia variedad de sistemas operativos y software, y son altamente escalables y configurables según tus necesidades.

**Pasos Detallados para una instancia EC2 en AWS:**

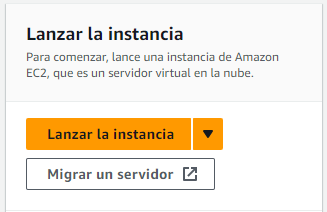
**Dirígete al Servicio EC2**

En la consola de AWS, busca y selecciona el servicio "EC2" en la sección "Compute".



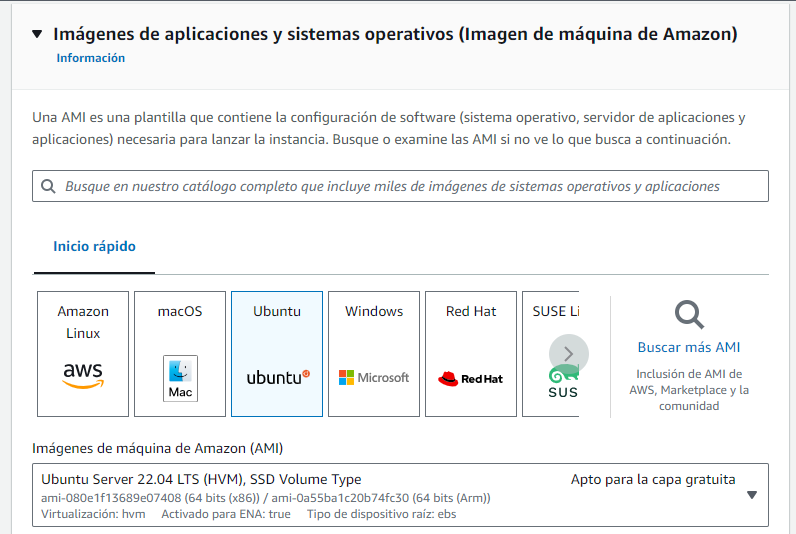
**Haz Clic en "Launch Instance"**

En el panel de control de EC2, haz clic en el botón "Launch Instance" para iniciar el proceso de lanzamiento de una nueva instancia.



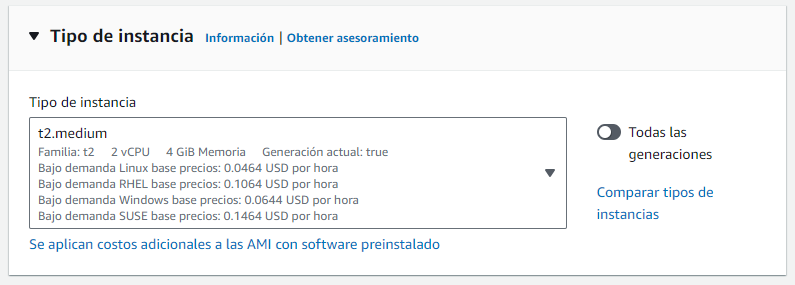
**Selecciona una Amazon Machine Image (AMI)**

En el primer paso del asistente de lanzamiento de instancias, elige una AMI que contenga el sistema operativo y el software que deseas utilizar en tu instancia. Puedes seleccionar una AMI prediseñada por AWS o una AMI personalizada.



**Elige un Tipo de Instancia**

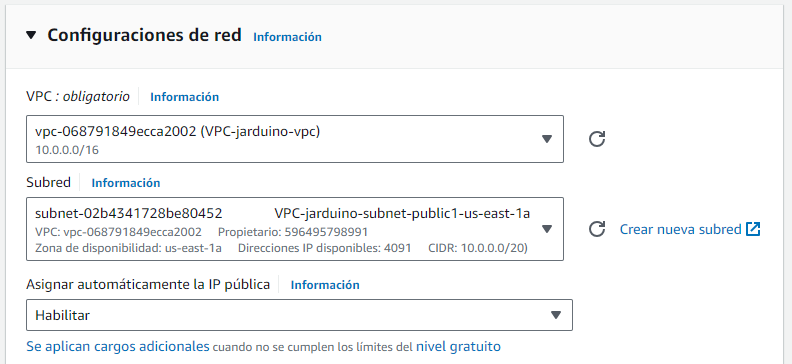
En este paso, selecciona el tipo de instancia que se ajuste a tus necesidades de capacidad de cómputo, memoria y almacenamiento. AWS ofrece una variedad de tipos de instancias, desde instancias de propósito general hasta instancias optimizadas para cómputo, memoria, almacenamiento y otros casos de uso específicos.



**Configura los Detalles de la Instancia**

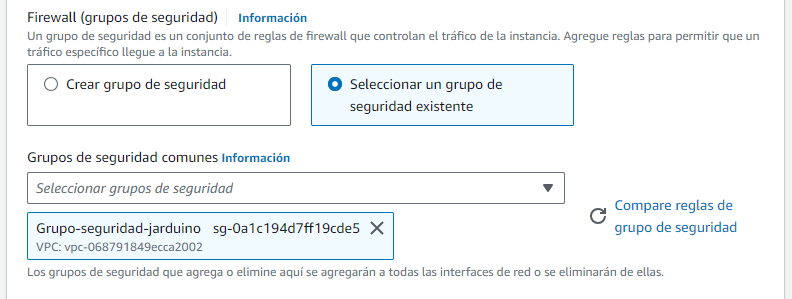
Aquí puedes configurar detalles como el número de instancias que deseas lanzar, la red en la que deseas lanzarlas, las subredes, las asignaciones de direcciones IP, etc.

Le indicamos la VPC que me hemos creado.



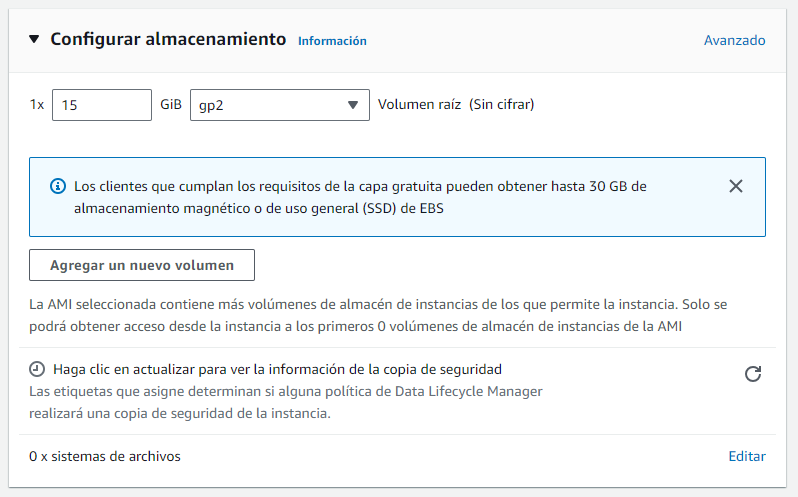
**Configura el Grupo de Seguridad**

Aquí puedes seleccionar un grupo de seguridad existente o crear uno nuevo para tu instancia EC2. Un grupo de seguridad actúa como un firewall virtual y te permite controlar el tráfico de red hacia y desde tu instancia, seleccionaremos el que hemos creado anteriormente.



**Añade Almacenamiento**

En este paso, puedes añadir y configurar el almacenamiento para tu instancia EC2. Puedes especificar el tamaño y el tipo de volumen, así como configurar opciones avanzadas como la encriptación.

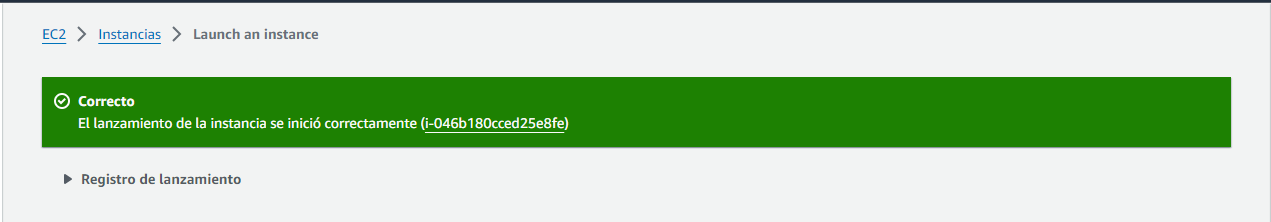


**Añade Etiquetas (Tags)**

Puedes añadir etiquetas a tu instancia para facilitar la organización y la gestión de tus recursos en AWS. Las etiquetas son pares de clave-valor que puedes utilizar para identificar y filtrar recursos en la consola de AWS.

**Revisa y Lanza la Instancia**

En la pantalla de revisión, asegúrate de que toda la configuración sea correcta y luego haz clic en "Launch" para lanzar tu instancia EC2.



### Configuración de un Servicio de Base de Datos Relacional (RDS)

### Transición de la Página Web a la Nube (EC2)

### Migración de la Base de Datos a la Nube (RDS)

## Verificación y Validación

### Pruebas de Funcionamiento en el Entorno Local y en la Nube

### Evaluación de la Seguridad y el Rendimiento

### Confirmación del Éxito del Proyecto

# ANÁLISIS DE RIESGOS

En este apartado se hará un análisis DAFO y de viabilidad económica del proyecto para el contexto propuesto y la potencial para otros contextos.

# BIBLIOGRAFÍA

En este apartado recogerás todas las referencias que hayas empleado para llevar a cabo tu proyecto: libros, recursos web, etc

# ANEXOS

En este apartado se pueden desarrollar con detalle todos los componentes o elementos del proyecto.

No es obligatorio, pero se recomienda su uso.