



INSTITUTO POLITÉCNICO NACIONAL

Escuela Superior de Cómputo

**Carrera:**

Ingeniería en Sistemas Computacionales

**Unidad de aprendizaje:**

Sistemas Distribuidos

**Práctica 9:**

Configuración de la nube.

**Alumno:**

Cardoso Osorio Atl Yosafat

Hernández Vázquez Jorge Daniel

**Profesor:**

Chadwick Carreto Arellano

**Fecha:**

10/06/2025

INSTITUTO POLITÉCNICO NACIONAL



## Índice de contenido

Antecedentes.....	5
Planteamiento del problema .....	6
Propuesta de solución .....	6
Materiales y métodos empleados.....	7
AWS .....	7
Materiales .....	7
Métodos Empleados .....	8
Google Cloud.....	9
Microsoft Azure.....	9
Servicios Principales Utilizados .....	9
Desarrollo .....	10
AWS .....	11
Principales servicios que ofrece AWS: .....	11
Configuración de una nube básica en AWS .....	13
Google Cloud.....	19
Principales servicios de Google Cloud:.....	20
Creación de cuenta en Google Cloud .....	21
Configuración de monitoreo en la nube .....	23
Creación de alertas de consumo .....	24
Configuración de facturación en la nube.....	26
Configuración de alertas de presupuesto .....	27
Configuración de Seguridad en la Nube: IAM en Google Cloud Platform (GCP) .....	28
Otorgamiento de Acceso a Usuarios .....	28
Configuración de servicios en la nube de Google Cloud Platform (GCP).....	29
Ejemplo de configuración de un servicio de despliegue: Cloud Run.....	30
Configuración de un servicio de bases de datos: Cloud SQL .....	34
Configuración de un servicio de almacenamiento: Cloud Storage.....	36
Microsoft Azure.....	39
Principales servicios que ofrece Azure:.....	39
Configuración de una nube básica en azure .....	41

Conclusión .....	73
Referencias .....	74

## Índice de figuras

Figura 1. Creación de cuenta AWS. ....	14
Figura 2. Configuración de EC2 en AWS. ....	15
Figura 3. Selección de tipo de instancias AWS. ....	15
Figura 4. Configuración de red en EC2 en AWS. ....	16
Figura 5. Configuración de almacenamiento en EC2 en AWS. ....	16
Figura 6. Finalización de la configuración de la instancia en AWS.....	17
Figura 7. Selección del motor de base de datos en AWS. ....	17
Figura 8. selección de la capa gratuita para base de datos en AWS.....	18
Figura 9. Configuración de credenciales en AWS. ....	18
Figura 10. Obtener datos de conexión en la base de datos de AWS. ....	19
Figura 11. Página de bienvenida de Google Cloud. ....	21
Figura 12. Crédito inicial de Google. ....	22
Figura 13. Formulario de Google Cloud. ....	22
Figura 14. Configuración de monitoreo en la nube. ....	23
Figura 15. Creación de alertas de consumo. ....	24
Figura 16. Configuración del disparador de alerta Google Cloud.....	25
Figura 17. Configuración de notificaciones en Google Cloud. ....	25
Figura 18. Finalización de alerta Google Cloud. ....	26
Figura 19. Facturación de la nube Google Cloud. ....	26
Figura 20. Configuración de alertas de presupuesto Google Cloud. ....	27
Figura 21. Configuración de seguridad Google Cloud. ....	28
Figura 22. Acceso a un usuario en Google Cloud. ....	29
Figura 23. Conexión con GitHub en Google Cloud. ....	30
Figura 24. Configuración de compilación Google Cloud. ....	31
Figura 25. Configuración de región del servicio Google Cloud. ....	31
Figura 26. Configuración de recursos Google Cloud. ....	32
Figura 27. Despliegue de aplicación Google Cloud. ....	33
Figura 28. Cloud SQL en Google Cloud. ....	34
Figura 29. Elección del motor de base de datos. ....	34
Figura 30. Configuración de la instancia de base de datos en Google Cloud. ....	35
Figura 31. Información de la instancia Google Cloud.....	35
Figura 32. Finalización de la instancia de base de datos de Google Cloud.....	36
Figura 33. Configuración de Cloud Storage en Google Cloud. ....	36
Figura 34. Nombre del bucket en Cloud Storage. ....	37
Figura 35. Selección de región en Cloud Storage. ....	37

Figura 36. Selección del modo de almacenamiento de archivos.....	38
Figura 37. Configuración de acceso Google Cloud.....	38
Figura 38. Políticas de seguridad Cloud Storage.....	39
Figura 39. Página de bienvenida de Microsoft Azure.....	42
Figura 40. Pagina de registro de informacion del perfil.....	42
Figura 41. Pagina de inicio de Azure, una vez creada una cuenta.....	43
Figura 42. Apartado donde se pueden ver los servicios de azure.....	44
Figura 43. Búsqueda desde Azure del servicio.....	45
Figura 44. Pagina principal de Azure monitor.....	45
Figura 45. Panel de Azure Insights   Applications Overview.....	46
Figura 46. Panel de control de alertas en azure monitor.....	47
Figura 47. Creacion de un tipo de alerta en azure monitor.....	47
Figura 48. Gráficas de Métricas en Azure Monitor.....	48
Figura 49. Pagina para la creación de un container registries.....	49
Figura 50. Detalles del container registry.....	50
Figura 51. Imágenes subidas a la nube en el container registry.....	51
Figura 52. Creación de un App Services en Azure.....	52
Figura 53. Detalles del app services.....	53
Figura 54. Configuracion de las imágenes del contenedor.....	55
Figura 55. Creación de un servidor de BD MySQL en Azure.....	56
Figura 56. Configuracion de la BD MySQL en Azure.....	57
Figura 57. Configuracion de la BD MySQL en Azure.....	58
Figura 58. Conexión a la BD desde MySQL Workbench.....	59
Figura 59. Creación de una cuenta de almacenamiento.....	59
Figura 60. Tipos de cuenta de alamancenamiento.....	60
Figura 61. Configuracion de seguridad.....	61
Figura 62. Configuracion de redes.....	62
Figura 63. Configuracion de proteccion de datoss.....	63
Figura 64. Configuracion de cifrado.....	64
Figura 65. Cifrado usado generalmente en los servicios de Azure.....	65
Figura 66. Configuracion de redes en el container registry.....	66
Figura 67. Proteccion de Microsoft Defender Cloud.....	66
Figura 68. Configuracion de redes en App Services.....	67
Figura 69. Proteccion de Microsoft Defender Cloud para app services.....	68
Figura 70. Certificados utilizados en el App Services.....	69
Figura 71. Control de acceso (IAM).....	70
Figura 72. Configuracion de redes para la BD en Azure.....	71
Figura 73. Apartado de copias de seguridad y restauracion de la BD.....	71
Figura 74. Proteccion de Microsoft Defender Cloud para la BD.....	72

## Antecedentes

Antes de la llegada de la computación en la nube, las empresas y organizaciones dependían de infraestructuras físicas locales (on-premise), lo que representaba diversas desventajas. Uno de los principales inconvenientes era los altos costos iniciales, ya que las organizaciones debían realizar inversiones significativas en servidores, licencias de software y en el mantenimiento de sus infraestructuras. Además, la escalabilidad limitada era otro desafío importante, ya que adaptarse a picos de demanda requería la compra de más hardware, lo que no siempre era práctico ni rentable. Además, este modelo de gestión implicaba una complejidad operativa, pues era necesario contar con equipos de TI dedicados exclusivamente a gestionar el hardware y las redes.

El cambio hacia la computación en la nube se aceleró con la globalización y el aumento de los datos, conocidos como big data, durante la década de 2000. Este avance dio paso a un modelo más flexible y eficiente, que ofrecía múltiples beneficios. Entre estos destacan el pago por uso, lo que eliminó la necesidad de realizar inversiones iniciales en hardware costoso; la escalabilidad bajo demanda, que permite ajustar los recursos en tiempo real según las necesidades de la empresa; y el acceso global, que proporciona a las empresas la posibilidad de acceder a servicios desde cualquier lugar del mundo, facilitando la expansión internacional.

Hoy en día, los tres proveedores más relevantes en el mercado de la computación en la nube son Amazon Web Services (AWS), Microsoft Azure y Google Cloud Platform (GCP). AWS, lanzado en 2006, comenzó con su primer servicio, S3, para escalar su infraestructura y soportar las ventas masivas. Actualmente, es el líder del mercado, con un 33% de participación. Por su parte, Microsoft Azure, iniciado en 2010 como Windows Azure, surgió para modernizar el modelo de negocio de Microsoft frente al auge de los servicios SaaS. Actualmente, ocupa el 22% del mercado. Finalmente, Google Cloud Platform, que comenzó en 2008 y se hizo público en 2011, surgió de la necesidad de Google de gestionar su infraestructura global y ofrecerla como un servicio. GCP cuenta actualmente con un 10% del mercado.

Este cambio hacia la nube ha propiciado una verdadera transformación digital, ya que el 94% de las empresas utilizan al menos un servicio en la nube. Además, han surgido nuevos paradigmas de gestión como el serverless, que permite la ejecución de código sin la necesidad de gestionar servidores, como en los casos de AWS Lambda o Google Cloud Functions. El uso de contenedores también ha ganado protagonismo, y Kubernetes, desarrollado originalmente por Google, se ha convertido en el estándar para la orquestación de contenedores. Por último, la inteligencia artificial accesible ha permitido democratizar el uso de machine learning, con servicios como AWS SageMaker, Azure AI y Vertex AI, que facilitan la implementación de estas tecnologías en empresas de todos los tamaños.

## Planteamiento del problema

Hoy en día, muchas aplicaciones web y servicios digitales enfrentan el reto de mantener un rendimiento óptimo y una alta disponibilidad a medida que crece el número de usuarios y las demandas de nuevas funcionalidades. Las arquitecturas monolíticas tradicionales presentan restricciones en términos de escalabilidad, mantenimiento y flexibilidad, lo que dificulta la rápida implementación de nuevas características y la adaptación a los cambios tecnológicos.

Además, el despliegue y la gestión de aplicaciones en servidores físicos o infraestructuras locales generan costos elevados, requieren mantenimiento constante y carecen de la elasticidad necesaria para ajustarse dinámicamente a las variaciones de demanda. Este escenario provoca que las organizaciones enfrenten dificultades para garantizar la continuidad del servicio y la satisfacción de los usuarios.

Por otro lado, la creciente adopción de tecnologías en la nube ofrece soluciones que superan estas limitaciones, mediante el uso de servicios escalables, flexibles y con alta disponibilidad. No obstante, persiste una brecha en el conocimiento práctico sobre cómo diseñar, desarrollar, desplegar y aprovechar adecuadamente los recursos que ofrece la nube para lograr sistemas eficientes y seguros.

En este contexto, surge la necesidad de realizar una práctica que permita explorar y comprender las arquitecturas en la nube, trabajando con proveedores como AWS, Azure y GCP en diferentes aspectos relacionados con la configuración de redes y seguridad, con el fin de garantizar un acceso adecuado y una protección eficaz de los datos. Esta experiencia facilitará el aprendizaje aplicado de los conceptos clave en arquitecturas modernas, y nos preparará para afrontar los desafíos reales en la creación y mantenimiento de aplicaciones escalables y resilientes.

## Propuesta de solución

Durante esta práctica de sistemas distribuidos, trabajaremos en el diseño y desarrollo de una solución que aborda los desafíos comunes en el desarrollo de aplicaciones escalables y resilientes, aprovechando los servicios en la nube de proveedores como AWS, Azure y Google Cloud Platform (GCP). Nuestro objetivo es mejorar la disponibilidad, el rendimiento y la seguridad de las aplicaciones, al mismo tiempo que optimizo los costos y el esfuerzo de mantenimiento.

**Seguridad:** Gestión de identidades (IAM), políticas de acceso, encriptación y protección de datos.

**Facturación:** Control de costos, presupuestos, alertas y optimización de recursos.

**Monitoreo:** Herramientas para supervisar rendimiento, disponibilidad y logs.

**Servicios:** Configuración de recursos clave como computación, almacenamiento y bases de datos.

## Materiales y métodos empleados

Para llevar a cabo la práctica y desarrollarla se implementó la arquitectura de Microservicios para resolver el problema de la venta de boletos y gestión se emplearon las siguientes herramientas y métodos:

### AWS

#### Materiales

Para el desarrollo y despliegue en la nube, se utilizaron los siguientes materiales y herramientas:

#### Equipos y Software:

- Computadora personal con sistema operativo Windows 10/11.
- Conexión a Internet estable para acceder a los servicios en la nube.

#### Amazon Web Services (AWS):

- Instancia EC2 (Elastic Compute Cloud) con el sistema operativo Ubuntu Server 22.04.
- Par de claves (.pem) para el acceso remoto y seguro a través de SSH.
- Configuración de grupos de seguridad (Security Groups) para permitir tráfico HTTP (puerto 5000) y SSH (puerto 22).

#### Herramientas de Desarrollo:

- **Flask:** Framework de Python utilizado para construir el microservicio RESTful.
- **Python 3.x** y **pip:** Para la gestión de paquetes y ejecución del código.
- **Visual Studio Code** o cualquier editor de texto para escribir y desarrollar el código.
- **Git y GitHub:** Herramientas para la gestión de versiones y control de código fuente.
- **Terminal Bash / PowerShell:** Utilizados para la ejecución de comandos y la conexión remota con la instancia en la nube.

#### Herramienta Adicional (opcional):

- **Ngrok:** Utilizado para realizar tunelado HTTP durante los entornos de prueba.

## Métodos Empleados

### Configuración del entorno de desarrollo local:

- Se creó un entorno virtual en Python utilizando venv para garantizar el aislamiento de las dependencias específicas del proyecto.
- Se procedió a la instalación de los paquetes requeridos, destacando Flask, a través de pip.

### Desarrollo del microservicio:

- Se desarrolló un microservicio RESTful empleando Flask, el cual simula una base de datos en memoria para gestionar productos.
- Se implementaron las rutas necesarias para llevar a cabo operaciones CRUD (crear, leer, actualizar y eliminar).
- El archivo principal del microservicio fue denominado app.py.

### Configuración del repositorio Git:

- El proyecto fue inicializado como un repositorio Git y se subió a GitHub para su gestión.
- Se utilizaron los comandos git pull y git checkout en la instancia EC2 para mantener el código actualizado de manera remota.

### Despliegue en la nube con AWS EC2:

- Se lanzó una instancia EC2 desde la consola de AWS, utilizando una imagen del sistema operativo Ubuntu Server.
- Se accedió de forma remota a la instancia mediante SSH, utilizando una clave privada (.pem).
- Se instalaron las dependencias necesarias (Python, pip, Flask) en la instancia EC2.
- El microservicio fue ejecutado con el comando python3 app.py, configurando el host a 0.0.0.0 para permitir conexiones externas.

### Configuración de acceso a la aplicación:

- Se configuró el grupo de seguridad de la instancia EC2 para permitir el tráfico entrante a través del puerto 5000.



- Se verificó la IP pública de la instancia y se accedió al microservicio tanto desde un navegador web como utilizando la herramienta curl.

### Pruebas y validación:

- Se llevaron a cabo pruebas de conexión y funcionalidad para cada una de las rutas del microservicio (GET, POST, PUT, DELETE).
- Se observó la respuesta en formato JSON y se validó el comportamiento de la aplicación frente a diferentes tipos de peticiones.

## Google Cloud

Para la configuración en la nube, se utilizaron los siguientes servicios de Google Cloud Platform (GCP):

- **Google Cloud Monitoring:** Para la configuración del monitoreo de los servicios.
- **Facturación:** Para gestionar y monitorear los presupuestos del proyecto.
- **IAM (Identity and Access Management):** Para la configuración de la seguridad y acceso.
- **App Engine:** Para la implementación y configuración del servicio.
- **Cloud SQL y Compute Engine:** Para la configuración y gestión de la base de datos.
- **Cloud Storage:** Para la configuración de almacenamiento de los datos.

Además, se emplearon las siguientes herramientas de apoyo:

- **Gemini:** Asistente de inteligencia artificial utilizado para explicar secciones de uso específico.
- **GitHub:** Utilizado dentro del servicio de App Engine para la gestión del código.

## Microsoft Azure

### Servicios Principales Utilizados

#### a) Azure App Service – Aplicación Web

- Se configuró un servicio de tipo App Service (Web App) con sistema operativo Windows y tecnología Java 21, alineado con el lenguaje de desarrollo utilizado para las prácticas futuras.
- Se seleccionó el plan de hospedaje Free (F1) o Standard (S1), dependiendo de los requerimientos de rendimiento específicos.

- Se habilitó la opción de HTTPS obligatorio para garantizar la seguridad de las conexiones, además de configurar un entorno de ejecución compatible con la versión local utilizada durante el desarrollo.

#### **b) Azure Authentication (Autenticación y Autorización)**

- Se integró el módulo de Autenticación y Autorización de Azure App Service para asegurar y proteger el acceso a la aplicación, garantizando que solo los usuarios autorizados pudieran acceder a ella.

#### **c) Azure Monitor – Supervisión**

- Se activó Azure Monitor para recopilar métricas clave de rendimiento, incluyendo el uso de CPU, memoria, disponibilidad y tiempos de respuesta de la aplicación.
- Se habilitaron los logs de diagnóstico y la integración con Application Insights, lo que permitió rastrear errores y analizar el tráfico de la aplicación en tiempo real.
- Se configuraron alertas automáticas para notificar sobre fallos en el sistema o el sobreuso de recursos, garantizando una respuesta rápida ante posibles problemas.

#### **d) Cuenta de Almacenamiento de Azure**

- Se implementó una cuenta de almacenamiento del tipo StorageV2, adecuada para las necesidades de la aplicación.
- Se utilizó Blob Storage para almacenar archivos estáticos, como imágenes, documentos y otros archivos generados por la aplicación, asegurando una gestión eficiente de los datos.

#### **e) Etiquetado y Organización de Recursos**

- Todos los recursos creados fueron organizados en un grupo de recursos lógico dentro de Azure, facilitando su administración.
- Se aplicaron etiquetas clave-valor (tags) para facilitar la clasificación y gestión de los recursos según su proyecto, entorno y responsable, mejorando la eficiencia en la administración de los mismos.

## **Desarrollo**

En esta práctica se desarrolló una Aplicación Web Progresiva (PWA) para gestionar la venta de boletos para un cine, utilizando tecnologías modernas como HTML, CSS, JavaScript, y Node.js para el backend, junto con MySQL para la base de datos. El propósito principal de

esta PWA es ofrecer una experiencia de usuario fluida y accesible, permitiendo la compra de boletos en tiempo real, notificaciones de disponibilidad de cartelera y el correcto funcionamiento incluso sin conexión a internet, es decir la posibilidad de poder descargar la PWA.

## AWS

AWS (Amazon Web Services) es una plataforma de servicios en la nube que ofrece una amplia gama de soluciones informáticas a través de internet, proporcionadas por Amazon. AWS es una de las principales plataformas de servicios en la nube utilizadas por empresas y desarrolladores para almacenar, gestionar y analizar datos, así como para ejecutar aplicaciones sin necesidad de tener infraestructura física propia.

Principales servicios que ofrece AWS:

### ***Cómputo:***

- **Amazon EC2 (Elastic Compute Cloud):** Proporciona servidores virtuales para ejecutar aplicaciones en la nube.
- **AWS Lambda:** Permite ejecutar código sin necesidad de gestionar servidores.
- **Elastic Beanstalk:** Un servicio que facilita el despliegue y gestión de aplicaciones sin tener que manejar la infraestructura subyacente.
- **Amazon Lightsail:** Un servicio más simple para usuarios que no necesitan una infraestructura tan compleja.

### ***Almacenamiento:***

- **Amazon S3:** Es un servicio de almacenamiento de objetos que permite almacenar y recuperar cualquier cantidad de datos en cualquier momento.
- **Amazon EBS:** Proporciona almacenamiento de bloques para instancias EC2, ideal para bases de datos y aplicaciones que requieren acceso rápido y persistente.
- **Amazon Glacier:** Almacenamiento a largo plazo con baja frecuencia de acceso, ideal para archivar datos.

### ***Bases de datos:***

- **Amazon RDS:** Ofrece bases de datos relacionales gestionadas (como MySQL, PostgreSQL, MariaDB, Oracle, SQL Server).
- **Amazon DynamoDB:** Una base de datos NoSQL completamente gestionada.

- **Amazon Aurora:** Es una base de datos relacional compatible con MySQL y PostgreSQL, pero con un rendimiento mejorado.

#### ***Redes:***

- **Amazon VPC (Virtual Private Cloud):** Permite crear una red privada dentro de AWS para configurar y controlar el acceso a recursos de manera segura.
- **Amazon Route 53:** Es un servicio de DNS que gestiona el enrutamiento de tráfico web.
- **AWS Direct Connect:** Establece una conexión de red dedicada entre tu infraestructura local y AWS.

#### ***Herramientas de desarrollo:***

- **AWS CodeBuild:** Es un servicio que compila código fuente, realiza pruebas y genera artefactos listos para desplegar.
- **AWS CodeDeploy:** Ayuda a automatizar el despliegue de aplicaciones en instancias EC2 o en otros servicios.
- **AWS CodePipeline:** Permite crear un flujo de trabajo de CI/CD, es decir de integración continua/despliegue continuo.

#### ***Inteligencia artificial y aprendizaje automático:***

- **Amazon SageMaker:** Es un servicio completamente gestionado que permite crear, entrenar y desplegar modelos de machine learning.
- **AWS Rekognition:** Análisis de imágenes y vídeos para identificar objetos, escenas y actividades.
- **AWS Polly:** Convierte texto en habla natural.
- **AWS Lex:** Crea interfaces de conversación mediante chatbots.

#### ***Seguridad, identidad y cumplimiento:***

- **AWS Identity and Access Management (IAM):** Gestiona usuarios y permisos para controlar el acceso a los servicios y recursos de AWS.
- **Amazon GuardDuty:** Es un servicio de detección de amenazas para monitorear la seguridad en tiempo real.
- **AWS Shield:** Protege contra ataques DDoS.

### *Análisis de datos:*

- **Amazon EMR (Elastic MapReduce):** Procesa grandes volúmenes de datos usando Apache Hadoop, Spark y otros marcos.
- **Amazon Redshift:** Es un servicio de almacenamiento de datos que permite analizar grandes volúmenes de datos de forma rápida y económica.
- **AWS Kinesis:** Facilita la recopilación, procesamiento y análisis en tiempo real de grandes flujos de datos.

### *IoT (Internet de las cosas):*

- **AWS IoT Core:** Permite conectar dispositivos IoT a la nube de manera segura y gestionada.
- **AWS IoT Greengrass:** Lleva la inteligencia de AWS a dispositivos IoT para ejecutar funciones de computación local.

### *Monitoreo y gestión:*

- **Amazon CloudWatch:** Monitorea la infraestructura y las aplicaciones en tiempo real, proporcionando métricas y registros.
- **AWS CloudTrail:** Registra todas las actividades realizadas en tu cuenta de AWS para auditorías de seguridad y cumplimiento.

## Configuración de una nube básica en AWS

### *Creación de una cuenta de AWS*

Para crear una cuenta en AWS, lo primero que debemos hacer es ingresar a [aws.amazon.com](https://aws.amazon.com) y hacer clic en la opción "Crear una cuenta de AWS". Una vez en la página de registro, debemos proporcionar información de contacto como nombre, dirección de correo electrónico y detalles de pago, ya que AWS requiere una tarjeta de crédito válida para la verificación de la cuenta, aunque muchos de sus servicios tienen una capa gratuita que puedes utilizar al inicio.

Luego, nos pedirá verificar nuestra identidad mediante un proceso de verificación telefónica, que generalmente consiste en recibir un código en el teléfono que debemos ingresar en la página para continuar con el registro. Después de la verificación, podremos seleccionar un plan de soporte, que varía según nuestras necesidades, desde opciones gratuitas hasta planes de soporte técnico más avanzados.

Una vez creada la cuenta, es recomendable seguir ciertas buenas prácticas de seguridad. En lugar de usar la cuenta raíz para la administración diaria de AWS, es aconsejable crear usuarios a través de IAM (Identity and Access Management). De esta manera, podremos gestionar y asignar permisos específicos a cada usuario según sea necesario. Además, es fundamental aplicar el principio de privilegio mínimo, otorgando solo los permisos necesarios para que los usuarios realicen sus tareas.

Otra recomendación importante es organizar a los usuarios en grupos para gestionar los permisos de manera más eficiente. Finalmente, para mejorar la seguridad, debemos habilitar la autenticación multifactor (MFA) en nuestra cuenta de AWS, lo que añadirá una capa extra de protección, ya que requerirá un segundo factor (como un código generado por una aplicación de autenticación) además de la contraseña para acceder a la cuenta.

**Descubra los productos del nivel gratuito con una nueva cuenta de AWS.**

Si desea obtener más información, visite [aws.amazon.com/free](https://aws.amazon.com/free).



### Regístrese en AWS.

Dirección de correo electrónico del usuario raíz  
Se utiliza para la recuperación de cuentas y tal como se describe en [Aviso de privacidad de AWS](#)

Nombre de cuenta de AWS  
Elija un nombre para su cuenta. Puede cambiar este nombre en la configuración de la cuenta después de registrarse.

**Verificar la dirección de correo electrónico**

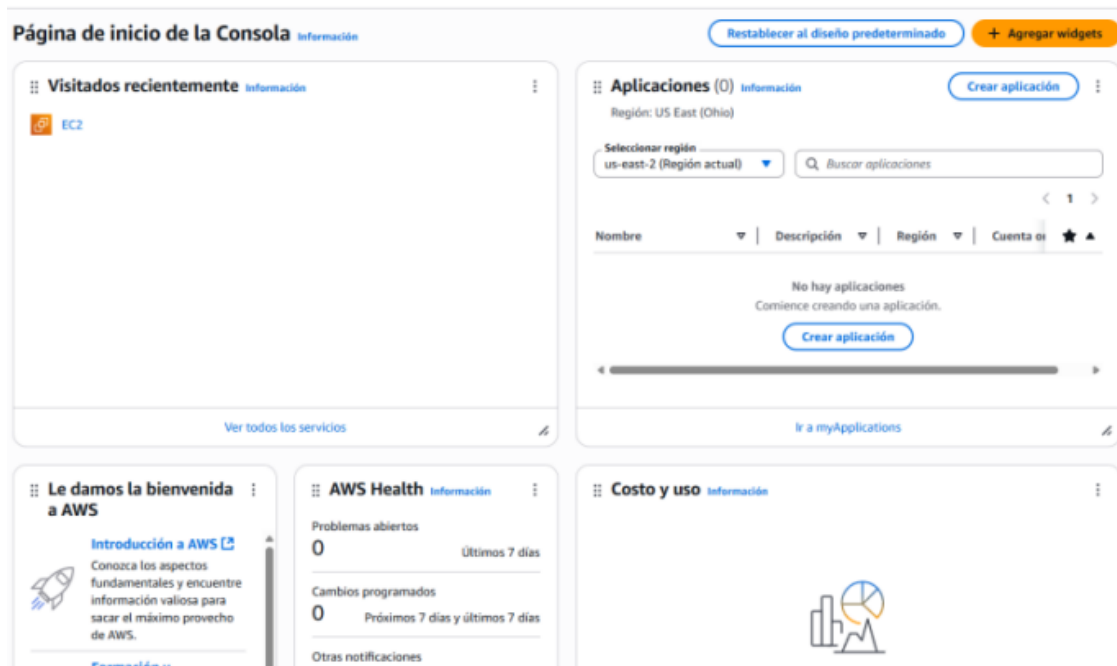
O

Iniciar sesión en una cuenta de AWS existente

**Figura 1. Creación de cuenta AWS.**

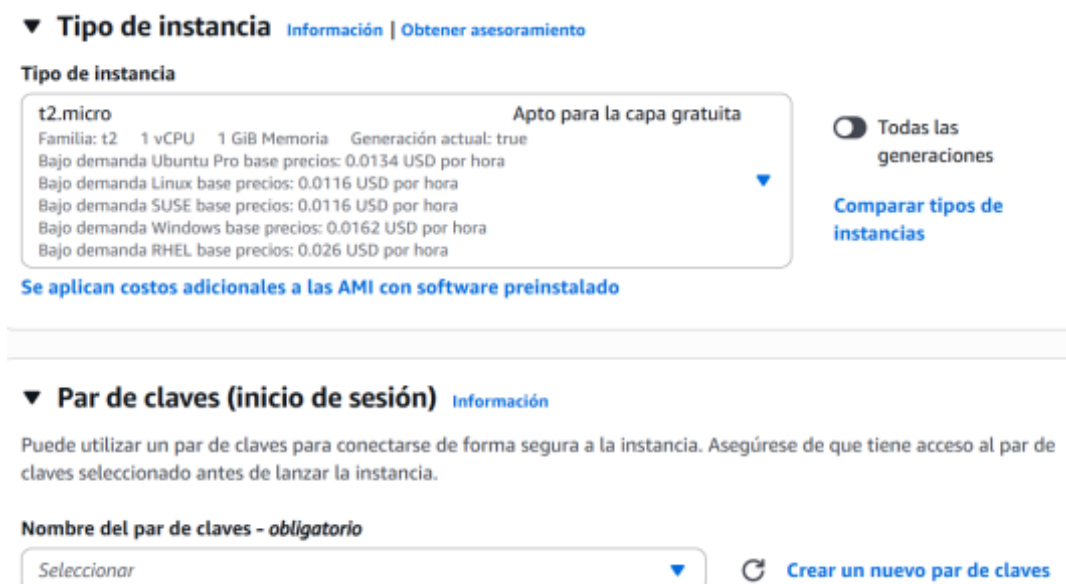
### *Configuración básica de EC2*

Para configurar una instancia básica en EC2, primero debemos acceder a la consola de EC2 de AWS. Una vez dentro, debemos hacer clic en la opción "Iniciar Instancia", que nos llevará al asistente de creación de instancias. En este paso, se nos pedirá seleccionar una Amazon Machine Image (AMI), que es una imagen preconfigurada que contiene el sistema operativo y las aplicaciones necesarias para el funcionamiento de la instancia. Es importante elegir la AMI adecuada según los requerimientos del proyecto.



**Figura 2. Configuración de EC2 en AWS.**

Luego, se nos presentará la opción de elegir un tipo de instancia que varíe según las necesidades de CPU y memoria que necesitemos para nuestra tarea. A continuación, deberemos configurar los detalles de red y almacenamiento, donde podemos definir la red en la que se ejecutará la instancia y la cantidad de almacenamiento que se asignará a la misma. Esto garantizará que la instancia tenga suficiente capacidad para funcionar de manera eficiente.



**Figura 3. Selección de tipo de instancias AWS.**

El siguiente paso es configurar los grupos de seguridad, que actúan como un firewall para controlar el acceso a la instancia. Aquí, podemos definir qué puertos estarán abiertos y qué direcciones IP podrán acceder a nuestra instancia.

▼ **Configuraciones de red** Información Editar

**Red** Información  
vpc-0c2b772010dbab535

**Subred** Información  
Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

**Asignar automáticamente la IP pública** Información  
Habilitar  
Se aplican cargos adicionales cuando no se cumplen los límites del nivel gratuito

**Firewall (grupos de seguridad)** Información  
Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

☒ Crear grupo de seguridad ☐ Seleccionar un grupo de seguridad existente

Crearemos un nuevo grupo de seguridad denominado "launch-wizard-5" con las siguientes reglas:

☒ Permitir el tráfico de SSH desde  
Ayuda a establecer conexión con la instancia

Cualquier lugar  
0.0.0.0/0

*Figura 4. Configuración de red en EC2 en AWS.*

▼ **Configurar almacenamiento** Información Avanzado

1x 8 GiB gp3 Volumen raíz, 3000 IOPS, No cifrado

Los clientes que cumplan los requisitos de la capa gratuita pueden obtener hasta 30 GB de almacenamiento magnético o de uso general (SSD) de EBS

Agregar un nuevo volumen

The selected AMI contains instance store volumes, however the instance does not allow any instance store volumes. None of the instance store volumes from the AMI will be accessible from the instance

Haga clic en actualizar para ver la información de la copia de seguridad  
Las etiquetas que asigne determinan si alguna política de Data Lifecycle Manager realizará una copia de seguridad de la instancia.



*Figura 5. Configuración de almacenamiento en EC2 en AWS.*

Finalmente, se nos pedirá lanzar la instancia con un par de claves para acceder a ella de manera segura mediante SSH. Este par de claves nos permitirá conectarnos de forma remota y administrar la instancia sin necesidad de ingresar una contraseña. Una vez completados estos pasos, la instancia estará lista para su uso.



#### ID de la instancia

 i-064afa9dc9f8451e9 (Ejemplo 2)

1. Abra un cliente SSH.
2. Localice el archivo de clave privada. La clave utilizada para lanzar esta instancia es antonio-ssh.pem
3. Ejecute este comando, si es necesario, para garantizar que la clave no se pueda ver públicamente.  
 `chmod 400 "antonio-ssh.pem"`
4. Conéctese a la instancia mediante su DNS público:  
 `ec2-3-129-25-198.us-east-2.compute.amazonaws.com`

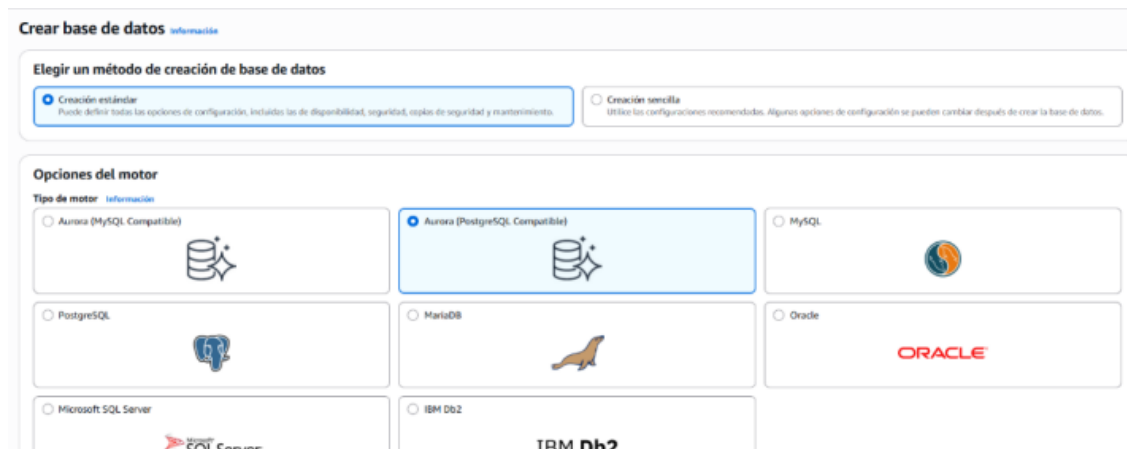
#### Ejemplo:

 `ssh -i "antonio-ssh.pem" ubuntu@ec2-3-129-25-198.us-east-2.compute.amazonaws.com`

**Figura 6. Finalización de la configuración de la instancia en AWS.**

#### Configuración básica AWS RDS y carga de una base de datos MySQL

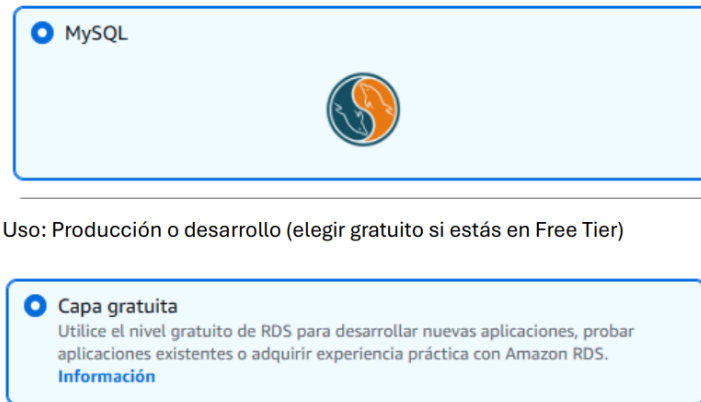
Para subir una base de datos MySQL a AWS RDS, lo primero que debemos hacer es crear una instancia de RDS en la consola de AWS. Accedemos a la consola de AWS desde el enlace: <https://console.aws.amazon.com/> y luego vamos a la sección de RDS. Una vez dentro, seleccionamos "Bases de datos" y hacemos clic en "Crear base de datos". En este paso, debemos elegir el motor de base de datos MySQL y seleccionar el tipo de uso que necesitamos, ya sea para producción o para desarrollo.



**Figura 7. Selección del motor de base de datos en AWS.**

Si estamos en el nivel Free Tier, podemos optar por la opción gratuita. A continuación, configuramos la instancia, definiendo un nombre para la base de datos, eligiendo un usuario administrador y una contraseña, seleccionando la versión de MySQL que queremos usar, y configurando el almacenamiento, que generalmente se puede dejar en 20 GB.

Motor: MySQL



The screenshot shows the AWS RDS console configuration page. At the top, 'Motor: MySQL' is displayed. Below it, a blue box contains the MySQL logo and the text 'MySQL'. Further down, another blue box is titled 'Capa gratuita' (Free Tier) and contains the text: 'Utilice el nivel gratuito de RDS para desarrollar nuevas aplicaciones, probar aplicaciones existentes o adquirir experiencia práctica con Amazon RDS.' and a link 'Información'.

Uso: Producción o desarrollo (elegir gratuito si estás en Free Tier)

**Figura 8. selección de la capa gratuita para base de datos en AWS.**

En la sección de conectividad, es importante habilitar el acceso público para poder conectarnos a la base de datos desde nuestra computadora. También debemos crear o seleccionar un grupo de seguridad (security group) que tenga abierto el puerto 3306, y asegurarnos de que permita conexiones desde nuestra IP. Después de completar esta configuración, podemos finalizar la creación de la instancia y esperar unos minutos hasta que se active.

#### ▼ Configuración de credenciales

##### Nombre de usuario maestro [Información](#)

Escriba un ID de inicio de sesión para el usuario maestro de la instancia de base de datos.

admin

1 a 16 caracteres alfanuméricos. El primer carácter debe ser una letra.

##### Administración de credenciales

Puede usar AWS Secrets Manager o administrar sus credenciales de usuario maestro.

☐

Administrado en AWS Secrets Manager - *más seguro*

RDS genera una contraseña y la administra durante todo su ciclo de vida mediante AWS Secrets Manager.

☒

Autoadministrado

Cree su propia contraseña o pida a RDS c

☐

Generar contraseña automáticamente

Amazon RDS puede generar una contraseña en su nombre, o bien puede especificar su propia contraseña.

##### Contraseña maestra [Información](#)

\*\*\*\*\*

##### Seguridad de la contraseña

Muy fuerte

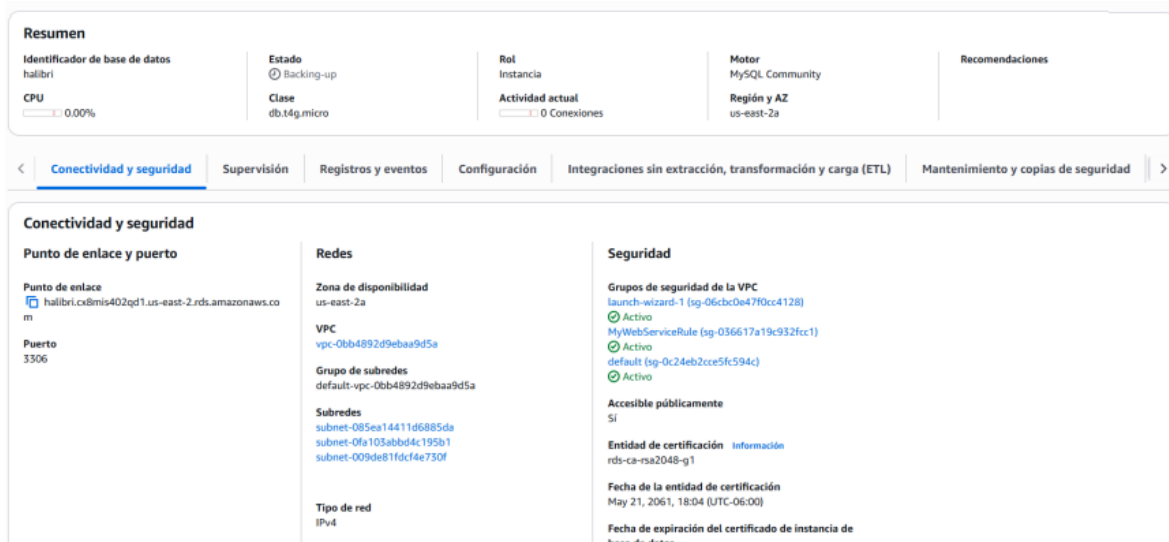
Restricciones mínimas: al menos 8 caracteres ASCII imprimibles. No puede contener ninguno de los siguientes símbolos: / ' \* @

##### Confirmar la contraseña maestra [Información](#)

\*\*\*\*\*

**Figura 9. Configuración de credenciales en AWS.**

Una vez que la instancia esté activa, necesitamos obtener los datos de conexión para poder interactuar con ella. Para esto, debemos ir nuevamente a la sección de RDS, seleccionar nuestra instancia recién creada y anotar los siguientes datos: el **Endpoint** (por ejemplo, database-1.xxxxxxxx.us-east-2.rds.amazonaws.com), el **Puerto** (que usualmente es 3306), y el **Usuario** que configuramos anteriormente.



**Figura 10. Obtener datos de conexión en la base de datos de AWS.**

Con los datos de conexión a la mano, ya podemos conectarnos a la base de datos desde nuestra computadora. Para ello, utilizamos el siguiente comando en MySQL, reemplazando <ENDPOINT> por el endpoint de nuestra base de datos, y <USUARIO> por el usuario administrador que configuramos:

```
mysql -h <ENDPOINT> -P 3306 -u <USUARIO> -p
```

Una vez conectados a la base de datos, podemos proceder a subir el archivo .sql que contiene la base de datos. Si tenemos un archivo llamado basededatos.sql, podemos cargarlo a la instancia de RDS con el siguiente comando:

```
mysql -h <ENDPOINT> -P 3306 -u <USUARIO> -p <basededatos.sql
```

Este comando cargará las tablas, los datos y los procedimientos almacenados desde el archivo .sql directamente a la base de datos en RDS. Finalmente, es importante verificar que los datos se hayan subido correctamente revisando las tablas y la información contenida en la base de datos.

Con estos pasos, habremos subido nuestra base de datos MySQL a AWS RDS y estaremos listos para empezar a trabajar con ella.

## Google Cloud

Google Cloud es una plataforma de servicios en la nube proporcionada por Google, que ofrece una amplia gama de herramientas y soluciones para empresas y desarrolladores. Estos servicios permiten a las organizaciones almacenar datos, analizar grandes volúmenes de información, crear aplicaciones, gestionar infraestructuras, y mucho más, todo ello de manera flexible y escalable.

Principales servicios de Google Cloud:

**Google Compute Engine (GCE):** Ofrece máquinas virtuales (VMs) que se pueden usar para ejecutar aplicaciones y servicios. Es una plataforma de infraestructura como servicio (IaaS) que permite gestionar servidores virtuales en la nube.

**Google Kubernetes Engine (GKE):** Servicio de contenedores basado en Kubernetes que facilita la implementación y administración de aplicaciones en contenedores a gran escala.

**Google App Engine (GAE):** Una plataforma como servicio (PaaS) que permite a los desarrolladores construir y desplegar aplicaciones sin preocuparse por la infraestructura subyacente.

**Google Cloud Storage:** Almacenamiento de objetos para guardar y acceder a cualquier cantidad de datos, desde imágenes hasta archivos grandes de manera segura.

**BigQuery:** Un servicio de análisis de datos en la nube que permite realizar consultas SQL a grandes volúmenes de datos rápidamente.

**Google Cloud SQL:** Un servicio de base de datos gestionado que soporta bases de datos SQL como MySQL, PostgreSQL y SQL Server.

**Google Cloud Functions:** Funcionalidad sin servidor (serverless) para ejecutar fragmentos de código en respuesta a eventos, sin necesidad de gestionar servidores.

**Google Cloud Firestore:** Base de datos NoSQL gestionada en tiempo real para aplicaciones web y móviles.

**Google Cloud Pub/Sub:** Servicio de mensajería en tiempo real para la transmisión de eventos entre aplicaciones y servicios.

**Google Cloud Identity & Access Management (IAM):** Permite gestionar quién tiene acceso a qué recursos en Google Cloud.

**Google Cloud Spanner:** Una base de datos relacional de alta disponibilidad, escalable y distribuida globalmente.

**Google Cloud AI y Machine Learning:** Ofrece herramientas y APIs de inteligencia artificial y aprendizaje automático, como Google Vision AI, Natural Language API, y AutoML, para crear soluciones inteligentes.

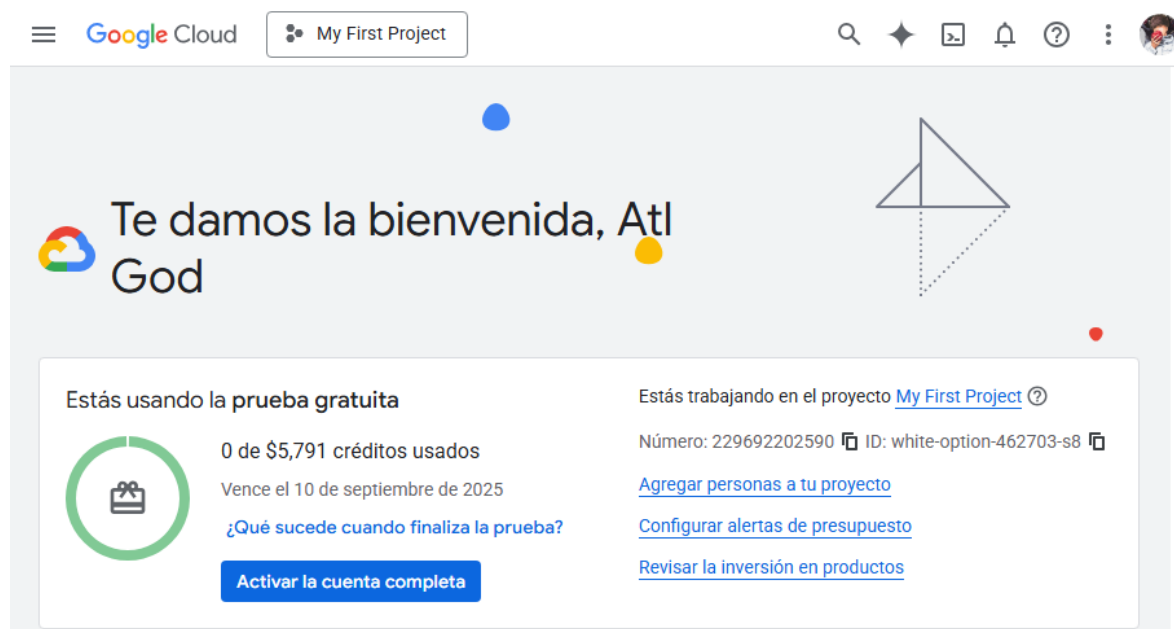
**Google Cloud CDN:** Red de entrega de contenido que acelera la entrega de contenido web y aplicaciones a través de una infraestructura global.

**Google Cloud Networking:** Soluciones de red como Cloud Load Balancing, Cloud VPN, y Cloud Interconnect que permiten gestionar el tráfico de red y conectar infraestructuras en la nube con redes locales.

**Google Cloud Monitoring y Logging:** Herramientas como Cloud Monitoring y Cloud Logging permiten supervisar el rendimiento de aplicaciones y servicios, así como realizar auditorías a los registros generados.

Creación de cuenta en Google Cloud

Para comenzar el proceso, se debe acceder al siguiente enlace: <https://console.cloud.google.com/>.



**Figura 11. Página de bienvenida de Google Cloud.**

En esta etapa, se solicitará iniciar sesión con una cuenta de correo de Google o Gmail. Inicialmente, Google ofrece un crédito gratuito de \$300, válido durante los próximos 90 días, lo que nos permitirá realizar la práctica. Para acceder a este beneficio, es necesario completar y confirmar la información relacionada con los pagos e impuestos de la cuenta.

## Acceso a todos los productos de Google Cloud

Obtén todo lo que necesitas para compilar y ejecutar tus apps, sitios web y servicios, incluidos Firebase y la API de Google Maps.

## \$300 en crédito gratuito

Prueba Google Cloud con \$300 de crédito para gastar en los próximos 90 días.

## Sin cargos automáticos

Solo comenzarás a pagar si decides activar una cuenta completa de pago por uso o si eliges pagar por adelantado. Conservarás el crédito gratuito restante.

*Figura 12. Crédito inicial de Google.*

Después de llenar el formulario, accederemos a la página principal donde podremos verificar que estamos dentro de la prueba gratuita y configurar la nube de Google.

## Paso 1 de 2 Información de la cuenta



Cardoso Osorio Atl Yosafat  
atl1cardoso0@gmail.com

[Cambiar de cuenta](#)

País

México

Cuando usas esta aplicación, aceptas las Condiciones del Servicio de [Google Cloud Platform](#), [la Prueba gratuita complementaria](#) y [cualquier servicio y APIs aplicables](#).

[Aceptar y continuar](#)

*Figura 13. Formulario de Google Cloud.*

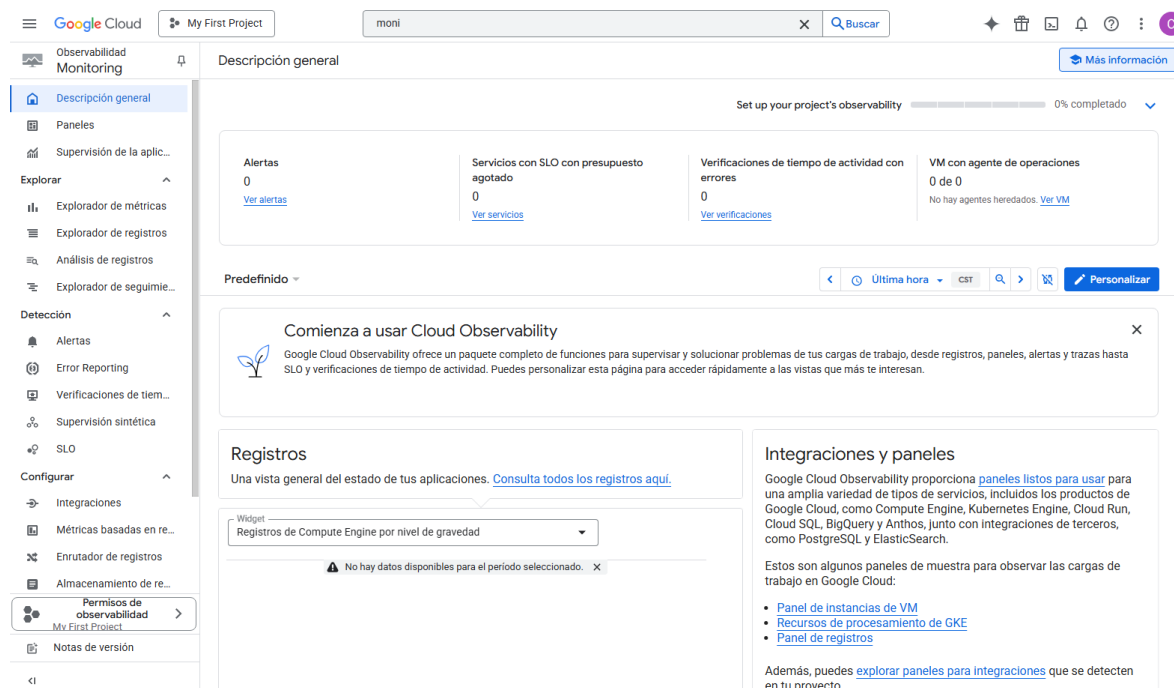
## Configuración de monitoreo en la nube

Google Cloud Platform (GCP) ofrece una amplia variedad de servicios para gestionar soluciones en la nube, y uno de los más destacados es Google Cloud Monitoring. Este servicio, accesible desde el menú lateral izquierdo de la plataforma, está diseñado específicamente para supervisar el rendimiento, la disponibilidad y el estado de las aplicaciones, infraestructura y servicios en la nube de manera integral.

Entre las principales funcionalidades de Google Cloud Monitoring se incluyen:

- Recopilación de métricas de los servicios (CPU, memoria, tráfico de red, latencia).
- Visualización interactiva de métricas mediante gráficos personalizables.
- Configuración de alertas basadas en umbrales específicos.
- Verificación de la disponibilidad de los endpoints desde diversas ubicaciones globales.
- Registro de errores para realizar un debugging rápido.

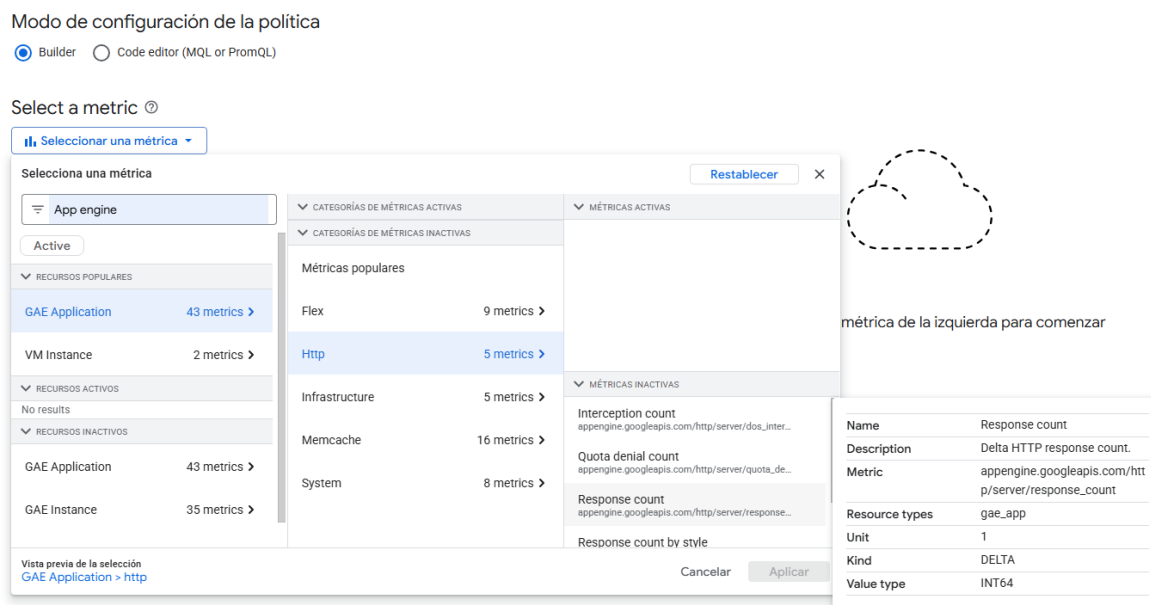
Es relevante señalar que las métricas estándar proporcionadas por Google Cloud Monitoring son gratuitas, lo que permite a los usuarios monitorizar sus recursos de manera básica sin incurrir en costos adicionales. No obstante, si se requiere acceder a métricas avanzadas, registros detallados o alertas personalizadas, se aplicarán cargos según el volumen de uso y la configuración de los recursos.



**Figura 14. Configuración de monitoreo en la nube.**

## Creación de alertas de consumo

Para configurar alertas en Google Cloud Monitoring, debemos acceder a la sección de alertas, donde se mostrará un resumen del historial de alertas activadas y las políticas creadas. Si no existe ninguna alerta configurada, podemos proceder a crear una nueva. En esta sección, se nos ofrecen dos opciones para definir la política: Builder o Code Editor. Para simplificar el proceso, seleccionaremos Builder, ya que es la opción más accesible y fácil de usar.



**Figura 15. Creación de alertas de consumo.**

En esta interfaz, elegimos la métrica que queremos monitorear. Para ello, debemos especificar un servicio (que puede estar en uso o no), seguido de la categoría de la métrica y finalmente la métrica en sí. Por ejemplo, creamos una alerta para el servicio que utilizaremos más adelante para monitorear las respuestas HTTP de este servicio.

Una vez configurada la métrica, el siguiente paso es definir cuándo se activará la alerta. GCP ofrece varias opciones de condiciones:

1. **Umbral:** Si en un intervalo de tiempo determinado, la métrica supera o baja de un valor específico.
2. **Ausencia de métrica:** Si no se recibe información sobre la métrica en un tiempo determinado.
3. **Pronóstico:** Predice si la métrica alcanzará un valor por encima o por debajo de un umbral preestablecido.



**Configure alert trigger**

**Condition Types**

☐ **Threshold**  
Condition triggers if a time series rises above or falls below a value for a specific duration window

☒ **Metric absence**  
Condition triggers if any time series in the metric has no data for a specific duration window

☐ **Forecast** Versión preliminar  
Condition triggers if any timeseries in the metric is projected to cross the threshold in the near future.

Alert trigger  
Cualquier serie tempor... ▼

Tiempo de ausencia del activador \*  
6 h ▼  
Esto anulará la ventana progresiva.

Nombre de la condición \*  
Alerta1

Next

**Figura 16. Configuración del disparador de alerta Google Cloud.**

En este caso, configuramos la alerta para que se active si no se recibe información sobre la métrica seleccionada en una hora. A continuación, elegimos los canales de notificación, como Google Chat, Slack, Email, SMS, entre otros. También podemos personalizar el asunto de la notificación, la prioridad de la alerta y agregar etiquetas a la política. Finalmente, se presenta una vista general de toda la configuración de la alerta. De esta manera, habremos creado una alerta que nos ayudará a monitorear el servicio configurado. Podemos agregar más condiciones si lo deseamos, pero para esta práctica, solo utilizamos una alerta simple.

**Configure notifications and finalize alert**

**Configure notifications** Recommended

☒ Usar el canal de notificaciones

Canales de notificaciones  
Correo Electronico ▼

Asunto de la notificación  
Se realizo la alerta de App Engine

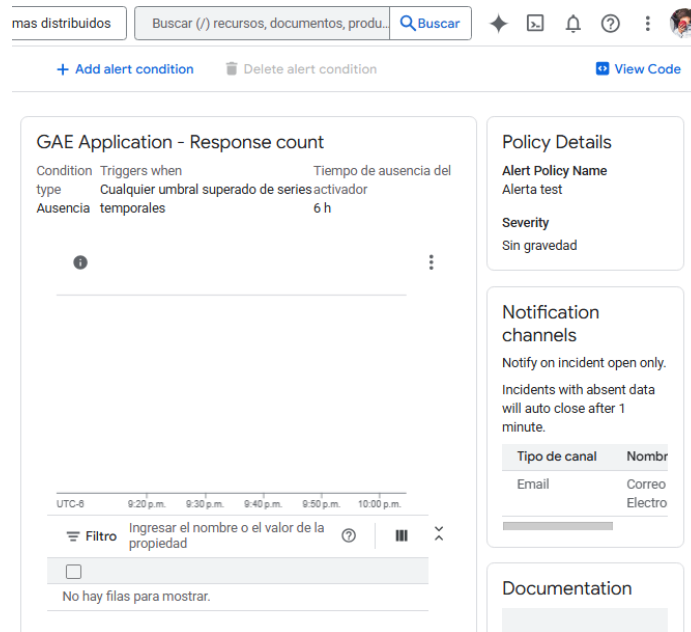
**i** We recommend that you create multiple notification channels for redundancy purposes. Google has no control of many of the delivery systems after we have passed the notification to that system. Additionally, a single Google service supports Cloud Console Mobile App, PagerDuty, Webhooks, and Slack. If you use one of these notification channels, then use email, SMS, or Pub/Sub as the redundant channel.

Más información [↗](#)

☐ Notify on incident closure

Incident autoclose duration ▼  
Si faltan datos, selecciona la duración después de la cual se cerrará automáticamente el incidente.

**Figura 17. Configuración de notificaciones en Google Cloud.**

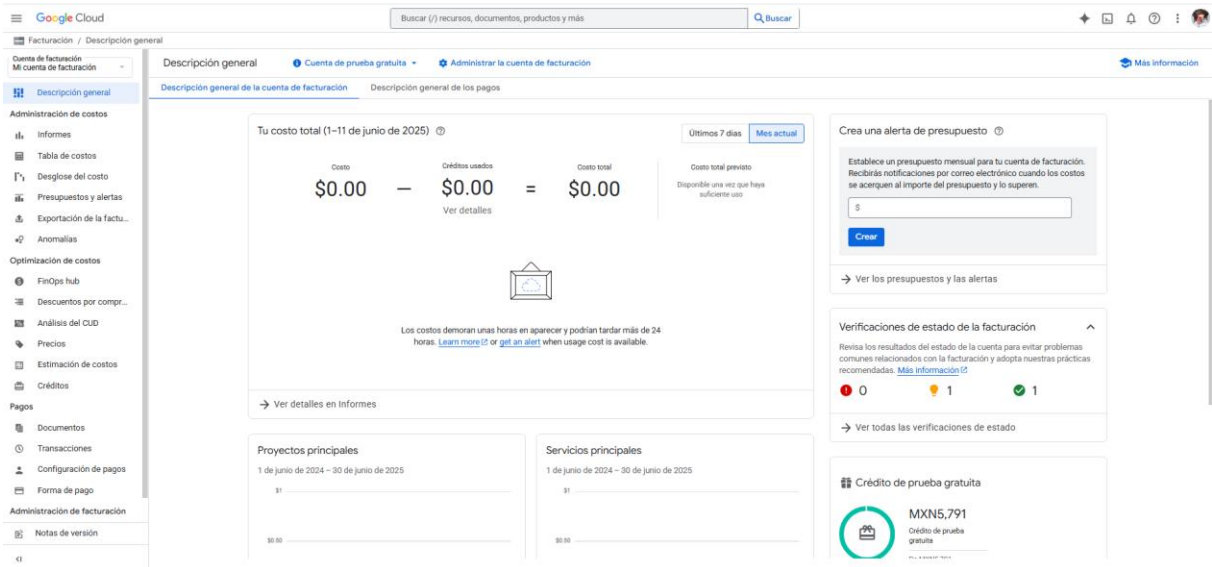


**Figura 18. Finalización de alerta Google Cloud.**

## Configuración de facturación en la nube

En cuanto a la facturación, GCP proporciona herramientas para controlar y optimizar los costos asociados con el uso de servicios en la nube. Estas herramientas incluyen:

- Modificación de datos fiscales.
- Monitoreo de gastos.
- Asignación de presupuestos.
- Configuración de métodos de pago.
- Realización de cotizaciones de servicios.



**Figura 19. Facturación de la nube Google Cloud.**

## Configuración de alertas de presupuesto

Dentro de la sección de Presupuestos y alertas, creamos un presupuesto y configuramos una alerta de consumo. Para ello, comenzamos a llenar los campos correspondientes, como:

- *Intervalo de tiempo de verificación*: 1 mes.
- *Proyectos*: seleccionamos el proyecto que estamos utilizando en esta práctica.
- *Servicios*: incluimos todos los servicios que utilizaremos.
- *Condiciones*: en caso de tener promociones activas, optamos por no filtrarlas, para poder ver todos los costos, incluyendo los servicios con promoción (en este caso, nuestra cuenta con una promoción).

Luego, especificamos el importe en dólares que será el límite de consumo, en este caso, 20 USD. Finalmente, configuramos las notificaciones para que se envíen alertas cuando se haya superado un porcentaje del presupuesto, en este caso a 50%, 90% y 100%. Podemos modificar estos valores y configurar quién recibirá las notificaciones, por lo que incluimos a todos los usuarios del proyecto. Así, habremos finalizado la configuración del presupuesto y las alertas de consumo.

**3 Acciones**

Set alert threshold rules

Envía notificaciones de alerta por correo electrónico luego de que el gasto real o el previsto superen un importe específico o cierto porcentaje del presupuesto. [Más información.](#)

Porcentaje del presup: 50 %	Importe 1 * \$ 10	Cuándo se activa 1 Real	
Porcentaje del presup: 90 %	Importe 2 * \$ 18	Cuándo se activa 2 Real	
Porcentaje del presup: 100 %	Importe 3 * \$ 20	Cuándo se activa 3 Real	

[+ Agregar limite](#)

Administra las notificaciones

☒ Alertas por correo electrónico a administradores de facturación y usuarios

☐ Alertas por correo electrónico a los propietarios del proyecto

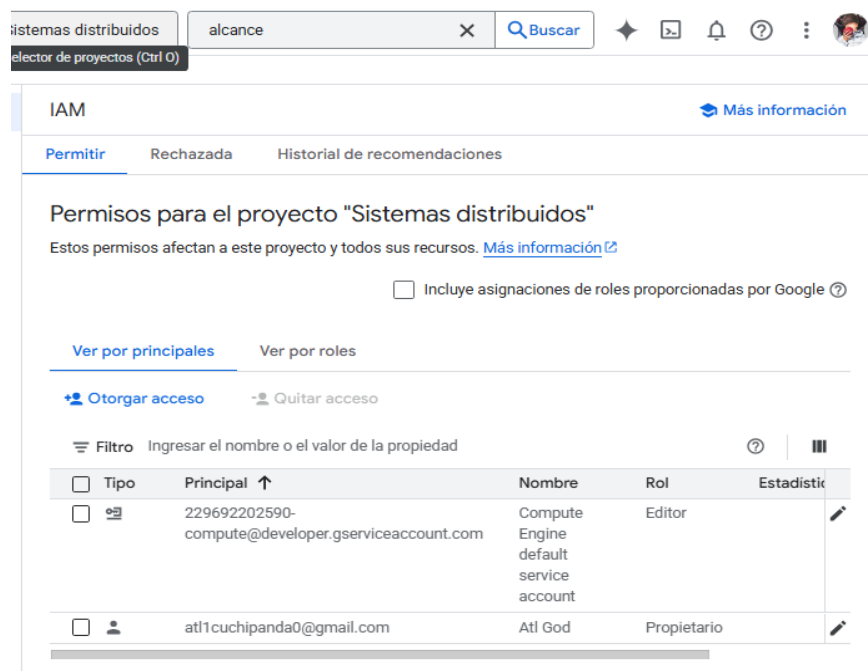
☐ Vincular los canales de notificación por correo electrónico de Monitoring a este presupuesto  
Selecciona un proyecto y un máximo de 5 canales de notificaciones por correo electrónico de Monitoring.

☐ Conectar un tema de Pub/Sub a este presupuesto  
Seleccionar un proyecto y un tema de Pub/Sub. Todos los que puedan ver este presupuesto también podrán ver el ID del proyecto y el nombre del tema. Es posible que no se pueda agregar un tema de Pub/Sub si pertenece a una organización que habilitó el [uso compartido restringido del dominio](#).

Figura 20. Configuración de alertas de presupuesto Google Cloud.

## Configuración de Seguridad en la Nube: IAM en Google Cloud Platform (GCP)

Google Cloud Platform (GCP) ofrece diversas herramientas de seguridad para proteger infraestructuras, datos y aplicaciones en la nube. Estas herramientas abarcan diferentes aspectos, desde la gestión de identidades hasta la defensa contra amenazas avanzadas. En este caso, nos enfocaremos en IAM (Identity and Access Management), un servicio esencial para gestionar los permisos y accesos dentro de la plataforma. IAM es accesible a través del menú lateral izquierdo en la consola de GCP, y su principal función es controlar quién puede acceder a los recursos y qué acciones pueden realizar en ellos.



**Figura 21. Configuración de seguridad Google Cloud.**

El objetivo principal de IAM es aplicar el principio de privilegios mínimos, lo que garantiza que los usuarios, servicios y grupos solo tengan los permisos necesarios para desempeñar sus tareas específicas. Este enfoque de seguridad ayuda a reducir riesgos al limitar las acciones que pueden llevar a cabo las personas o servicios en la nube, asegurando que no se otorguen permisos innecesarios que puedan comprometer la infraestructura.

### Otorgamiento de Acceso a Usuarios

En entornos colaborativos o cuando varios miembros de un equipo gestionan una misma infraestructura en la nube, es necesario conceder accesos a los usuarios de manera controlada. Para ello, IAM permite asignar roles a los usuarios mediante sus direcciones de correo electrónico. Estos roles vienen predefinidos con permisos específicos que definen qué acciones puede realizar cada usuario en los recursos de GCP.

Además de los roles predeterminados, IAM ofrece la opción de configurar condiciones adicionales de acceso. Estas condiciones pueden incluir restricciones basadas en factores

como el tiempo o el servicio específico al que se debe tener acceso. Por ejemplo, se pueden configurar restricciones para que un usuario solo pueda acceder a ciertos recursos en determinadas horas del día o en ciertos entornos de trabajo.

## Agregar principales

Las principales son usuarios, grupos, dominios o cuentas de servicio. [Más información sobre las principales de IAM](#)

Principales nuevas \*

atl1cardoso0@gmail.com X

danydrew76@gmail.com X

?

## Asignar roles

Los roles se componen de conjuntos de permisos y determinan lo que la principal puede hacer con este recurso. [Más información](#)

Rol \*

Acceso para depurar las VM ... ▼

Capacidad de leer o administrar instancias v2.

Condición de IAM (opcional) ?

+ Agregar condición de IAM



*Figura 22. Acceso a un usuario en Google Cloud.*

De esta forma, IAM permite gestionar de manera eficiente y segura los permisos en un entorno de nube, asegurando que cada usuario solo tenga acceso a los recursos y funcionalidades que realmente necesita para llevar a cabo su trabajo.

## Configuración de servicios en la nube de Google Cloud Platform (GCP)

Google Cloud Platform (GCP) ofrece una amplia gama de servicios, cada uno diseñado para cumplir con necesidades específicas, como la gestión de archivos, bases de datos, análisis, inteligencia artificial y despliegue de aplicaciones. A continuación, se comparan los servicios de despliegue más relevantes de GCP:

### Servicios de despliegue en GCP

**Cloud Run (Serverless):** Ideal para aplicaciones como APIs REST, microservicios o aplicaciones sin estado que necesitan escalar a cero, es decir, solo se paga cuando hay tráfico.

**Google Kubernetes Engine (GKE):** Perfecto para aplicaciones complejas con microservicios, que requieren alta disponibilidad y personalización avanzada.

**App Engine (PaaS):** Pensado para aplicaciones web tradicionales, permitiendo un despliegue simplificado.

**Compute Engine (VMs):** Recomendado para aplicaciones legadas o aquellas que requieren un control completo sobre el sistema operativo.

**Firebase Hosting + Cloud Functions:** Ideal para aplicaciones full-stack, combinando hosting y funciones en la nube.

#### Ejemplo de configuración de un servicio de despliegue: Cloud Run

Para desplegar una aplicación utilizando Cloud Run, el proceso comienza seleccionando el origen del proyecto, ya sea desde un Docker, GitHub o una función específica. En este caso, se elige GitHub. Para ello, primero configuramos el acceso de nuestra cuenta de GitHub con GCP y luego especificamos el repositorio y la rama donde se encuentra el proyecto.

1 Repositorio de código fuente

Proveedor del repositorio  
GitHub

Repositorio \*  
ATL1GOD/APPSCOM

¿No encuentras tu repositorio? [Administrar repositorios conectados](#)

☒ Entiendo que se transferirá el contenido de GitHub de los repositorios seleccionados a este proyecto de GCP para proporcionar el servicio conectado. Las principales que tengan acceso a este proyecto de GCP con permisos suficientes podrán crear y ejecutar activadores en estos repositorios en función del contenido transferido de GitHub. También comprendo que el contenido de todos los activadores de la app de GitHub del proyecto de GCP se podría transferir a GitHub a fin de proporcionar funcionalidades, como mostrar los nombres de los activadores en los resultados de la compilación de GitHub. Esta acción se aplicará a todos los activadores futuros y existentes de la app de GitHub en el proyecto. [Más información](#)

Los registros de compilación se enviarán a GitHub.

[Siguiente](#)

**Figura 23. Conexión con GitHub en Google Cloud.**

El siguiente paso consiste en seleccionar el método de compilación, que puede ser con un Dockerfile o mediante un lenguaje de programación específico. En este caso, se selecciona Node.js como lenguaje de programación. Además, se configura la región del servicio para optimizar la latencia.

✓ Repositorio de código fuente

2 Configuración de compilación

Rama \*

^main\$

Usa una expresión regular que coincida con una rama específica [Más información](#)

Coincide con la rama: main

Tipo de compilación

☐ Dockerfile

☒ Go, Node.js, Python, Java, .NET Core, Ruby o PHP, mediante los [paquetes de compilación de Google Cloud](#)

Directorio de contexto de compilación \*

/

El directorio se usará como contexto de compilación del paquete de compilación.

Punto de entrada

Comando para iniciar el servidor, déjalo en blanco a fin de utilizar el [comportamiento predeterminado](#)

Objetivo de la función

El nombre de la función exportada que se invocará. Deja el campo en blanco si el repositorio de código fuente es un servidor web.

Guardar

**Figura 24. Configuración de compilación Google Cloud.**

Luego, GCP asigna automáticamente una URL predeterminada para acceder al servicio. Se puede configurar el acceso a través del IAM (Identity and Access Management) y habilitar el acceso directo a internet. Se elige la facturación basada en solicitudes, y se ajusta el número de instancias mínimas para el servicio.

URL del extremo

<https://p6-sistemas-distribuidos-454091775059.northamerica-south1.run.app>

Autenticación \*

☒ Usar Cloud IAM para autenticar las solicitudes entrantes  
Cloud IAM autorizará todas las invocaciones del extremo de este servicio.

☒ Allow unauthenticated invocations  
Configures an IAM policy that allows access without a credential.

☐ Autenticación obligatoria  
Administrar los usuarios autorizados con Cloud IAM.

Facturación

☒ Basada en solicitudes  
Se cobra solo cuando se procesan solicitudes. La CPU está limitada fuera de las solicitudes.

☐ Basada en instancias  
Se cobra por todo el ciclo de vida de las instancias. CPU completa durante toda la vida útil de cada instancia.

Escalamiento de servicios

☒ Ajuste de escala automático

Número mínimo de instancias

1

☐ Escalamiento manual **Versión preliminar**

Ingress

☐ Interno  
Permite el tráfico de tu proyecto, la VPC compartida y el perímetro de los Controles del servicio de VPC. El tráfico de otro servicio de Cloud Run se debe enrutar a través de una VPC. Se aplican limitaciones. [Más información](#)

☒ Todos  
Permite el acceso directo a tu servicio desde Internet

**Figura 25. Configuración de región del servicio Google Cloud.**

En cuanto a la configuración de recursos, se determina la memoria y CPU necesarias para el servicio, y se puede configurar una verificación para asegurar la disponibilidad del servicio. La mayoría de las configuraciones de redes, almacenamiento y seguridad se dejan por defecto, ya que GCP maneja de manera eficiente estos aspectos.

^ Editar contenedor

Repositorio de origen  
http://github.com/LeoYeudiel/P6\_Sistemas-Distribuidos

Activador de Cloud Build  
Se creará un activador de Cloud Build a fin de compilar y, luego, implementar tu código automáticamente.

Puerto de contenedor  
8080  
Las solicitudes se enviarán al contenedor de este puerto. Recomendamos detectar en `SPORT`, en lugar de en este número específico.

Configuración Variables y secretos Activaciones de volúmenes

Nombre del contenedor: placeholder-1 Edit

Comando de contenedor  
Deja el campo en blanco para usar el comando de punto de entrada definido en la imagen de contenedor.

Argumentos de contenedor  
Argumentos pasados al comando del punto de entrada.

Recursos

Memoria 512 MiB  
Es la memoria para asignar a cada instancia de este contenedor

CPU 1  
Es la cantidad de CPU virtuales asignadas a cada instancia de este contenedor

☐ GPU  
La GPU no es compatible con la región seleccionada. Para usar la GPU, selecciona una [región admitida](#).

Verificaciones de estado ?  
+ Agregar verificación de estado

Listo

**Figura 26. Configuración de recursos Google Cloud.**

Una vez completada la configuración, GCP desplegará automáticamente la aplicación, y podrás acceder a ella a través de la URL asignada.






## General

Facturación	Basada en solicitudes
Aumento de CPU de inicio	Habilitada
Simultaneidad	80
Tiempo de espera de la solicitud	300 seconds
Entorno de ejecución	Predeterminada

## Ajuste de escala automático

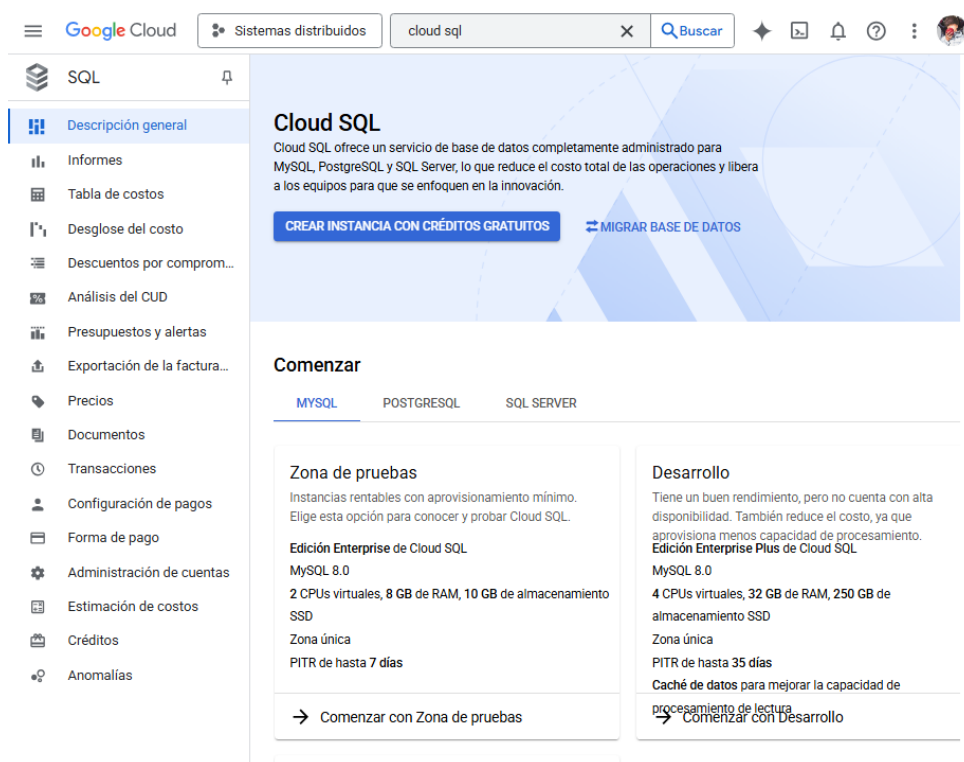
Cantidad mínima de instancias de revisión	1
Cantidad máxima de instancias de revisión	5

Imagen	<a href="https://gcr.io/cloudrun/placeholder@sha256:51a903e4c...">gcr.io/cloudrun/placeholder@sha256:51a903e4c...</a> 
Puerto	8080
Compilación	(no hay información disponible sobre la compilación) 
Fuente	(no hay información disponible sobre el origen) 
Comandos y argumentos	(punto de entrada del contenedor)
Límite de CPU	1

*Figura 27. Despliegue de aplicación Google Cloud.*

## Configuración de un servicio de bases de datos: Cloud SQL

Cloud SQL es un servicio de base de datos relacional de GCP que soporta MySQL, PostgreSQL y SQL Server.

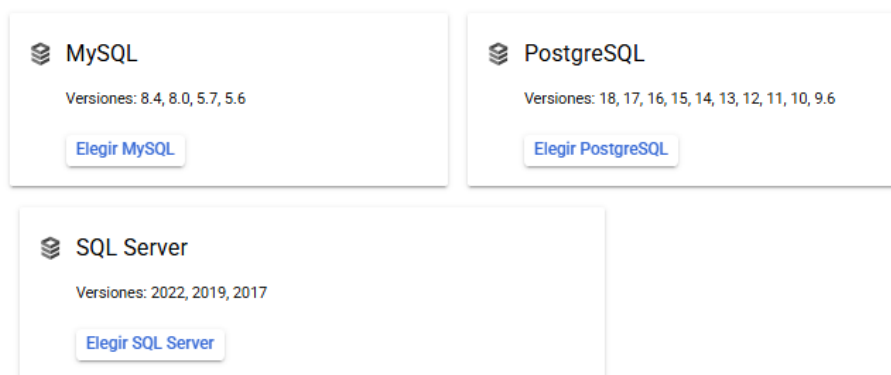


**Figura 28. Cloud SQL en Google Cloud.**

A continuación, se describe cómo crear una instancia de base de datos utilizando MySQL:

Primero, se selecciona el gestor de base de datos (en este caso, MySQL) y se habilita la API de Compute Engine para permitir la interacción con las instancias de máquinas virtuales.

### Elige tu motor de base de datos



**Figura 29. Elección del motor de base de datos.**

Luego, se configuran los detalles de la instancia, como el tipo de edición (según la disponibilidad, rendimiento y seguridad), el ID de la instancia y la contraseña del usuario raíz.

← Crea una instancia de MySQL

☐ Enterprise Plus

- ANS de disponibilidad del 99.99%
- Tiempo de inactividad por mantenimiento planificado en menos de un segundo
- Escalación vertical de instancias con tiempo de inactividad casi nulo
- Máquinas con rendimiento optimizado
- Hasta 35 días para la ventana de recuperación de un momento determinado
- Capacidad de procesamiento hasta 3 veces mayor con caché de datos
- Recuperación ante desastres avanzada con cambio fácil
- Compatibilidad con el grupo de conexiones administrado

☒ Enterprise

- ANS de disponibilidad del 99.95%
- Menos de 60 segundos de tiempo de inactividad por mantenimiento planificado
- Máquinas de uso general
- Hasta 7 días para la ventana de recuperación de un momento determinado

**Resumen**

Edición de Cloud SQL	Enterprise
Región	us-central1 (Iowa)
Versión de la base de datos	MySQL 8.0
CPU virtuales	2 CPU virtual(es)
RAM	8 GB
Caché de datos	Inhabilitada
Almacenamiento	10 GB SSD
Conexiones	IP pública
Copia de seguridad	Automatizada
Disponibilidad	Zona única
Recuperación de un momento determinado	Habilitada
Capacidad de procesamiento de la red (MB/s)	500 de 500
IOPS	Lectura: 6,300 de 15,000 Escritura: 6,300 de 15,000
Capacidad de procesamiento del disco (MB/s)	Lectura: 4.8 de 240.0 Escritura: 4.8 de 240.0

**Precio estimado (sin descuentos)**

Estos elementos representan únicamente los recursos de procesamiento, memoria y almacenamiento de Cloud SQL. No reflejan la configuración que estableciste para tu instancia. No se incluyen los descuentos en la factura.

[Más información](#)

Elemento	Precio
2 CPU virtuales (USD0.041 por CPU virtual por hora)	
8 GiB de RAM (USD0.007 por GiB por hora)	
10 GiB de SSD (USD0.17 por GiB al mes)	
Total sin descuentos por uso	

Elige un ajuste predeterminado para esta edición. Los ajustes predeterminados se pueden personalizar más adelante según sea necesario.

Ajuste predeterminado para esta edición: Se aplicó la configuración "Zona de pruebas"

Figura 30. Configuración de la instancia de base de datos en Google Cloud.

Se selecciona la región donde estará disponible la instancia y si se desea alta disponibilidad, se pueden habilitar varias zonas, lo que incrementa el costo.

### Información de la instancia

Versión de la base de datos \*  
MySQL 8.0

#### ✓ MOSTRAR VERSIONES SECUNDARIAS

ID de instancia \*  
sistemasdistribuidos  
Usa letras minúsculas, números y guiones. Comienza con una letra.

Contraseña \*  
•••••  
Establece una contraseña para el usuario raíz. [Más información](#)

☐ Sin contraseña

#### ✓ POLÍTICA DE CONTRASEÑAS

Figura 31. Información de la instancia Google Cloud.

Se configuran los recursos de la instancia, como la cantidad de CPU, RAM y almacenamiento, y se activa la opción de escalamiento automático para ajustarse a las necesidades de rendimiento.

También se puede definir cómo se conectará la instancia, ya sea por IP pública o privada, y se configuran las redes autorizadas para permitir el acceso.

Finalmente, se establece el plan de protección de datos, como las copias de seguridad automáticas, y se configura el mantenimiento periódico de la base de datos.

### Resumen

Edición de Cloud SQL	Enterprise
Región	us-central1 (Iowa)
Versión de la base de datos	MySQL 8.0
CPU virtuales	2 CPU virtual(es)
RAM	8 GB
Caché de datos	Inhabilitada
Almacenamiento	10 GB SSD
Conexiones	IP pública
Copia de seguridad	Automatizada
Disponibilidad	Zona única
Recuperación de un momento determinado	Habilitada
Capacidad de procesamiento de la red (MB/s)	500 de 500
IOPS	Lectura: 6,300 de 15,000 Escritura: 6,300 de 15,000
Capacidad de procesamiento del disco (MB/s)	Lectura: 4.8 de 240.0 Escritura: 4.8 de 240.0

### Precio estimado (sin descuentos)

Estos elementos representan únicamente los recursos de procesamiento, memoria y almacenamiento de Cloud SQL. No reflejan la configuración que estableciste para tu instancia. No se incluyen los descuentos en la factura.

[Más información](#)

Elemento	Precio
2 CPU virtuales	(USD0.041 por CPU virtual por hora)
8 GiB de RAM	(USD0.007 por GiB por hora)
10 GiB de SSD	(USD0.17 por GiB al mes)
<b>Total sin descuentos por uso</b>	

**Figura 32. Finalización de la instancia de base de datos de Google Cloud.**

Configuración de un servicio de almacenamiento: Cloud Storage

En GCP, Cloud Storage ofrece un almacenamiento de objetos escalable y duradero.

Sistemas distribuidos

selector de proyectos (Ctrl O)

Descripción general

Guías

¿Te sirvió esta página?

Te damos la bienvenida a Cloud Storage, Atl

Crear bucket

Ir a una ruta específica

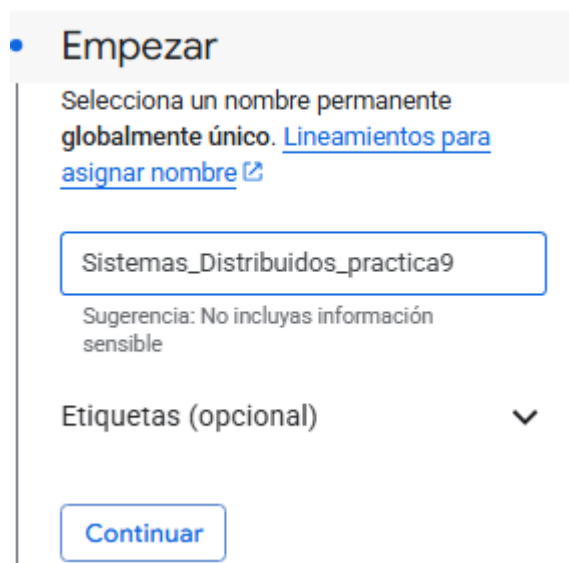
Ver lista de buckets

Retomar

**Figura 33. Configuración de Cloud Storage en Google Cloud.**

Los pasos para crear un bucket en Cloud Storage son los siguientes:

Se asigna un nombre único para el bucket, el cual debe ser globalmente único.



**Empezar**

Selecciona un nombre permanente globalmente único. [Lineamientos para asignar nombre](#)

Sistemas\_Distribuidos\_practica9

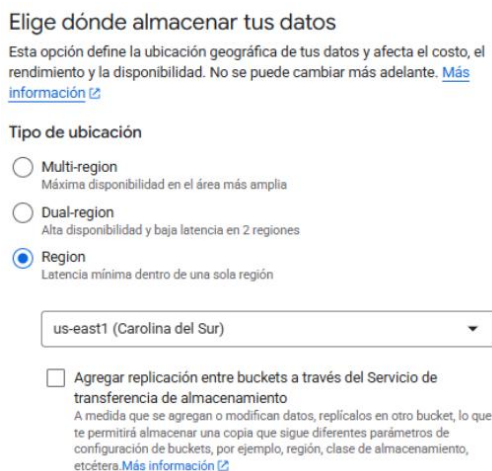
Sugerencia: No incluyas información sensible

Etiquetas (opcional) ▼

Continuar

**Figura 34. Nombre del bucket en Cloud Storage.**

Se selecciona la región donde se almacenarán los archivos. Esta puede ser una región única, una región dual o una opción multi-región para alta disponibilidad.



Elige dónde almacenar tus datos

Esta opción define la ubicación geográfica de tus datos y afecta el costo, el rendimiento y la disponibilidad. No se puede cambiar más adelante. [Más información](#)

Tipo de ubicación

☐ Multi-region  
Máxima disponibilidad en el área más amplia

☐ Dual-region  
Alta disponibilidad y baja latencia en 2 regiones

☒ Region  
Latencia mínima dentro de una sola región

us-east1 (Carolina del Sur)

☐ Agregar replicación entre buckets a través del Servicio de transferencia de almacenamiento

A medida que se agregan o modifican datos, repícalos en otro bucket, lo que te permitirá almacenar una copia que sigue diferentes parámetros de configuración de buckets, por ejemplo, región, clase de almacenamiento, etcétera. [Más información](#)

**Figura 35. Selección de región en Cloud Storage.**

Se define el modo de almacenamiento de los archivos, lo que afecta los costos y la velocidad de recuperación de los mismos.

### Elige cómo almacenar tus datos

Una clase de almacenamiento determina los costos del almacenamiento, la recuperación y las operaciones con diferencias mínimas en el tiempo de actividad. Elige si deseas administrar los objetos de forma automática o especifica una clase de almacenamiento predeterminada según la duración con la que planeas almacenar tus datos y tu carga de trabajo o caso de uso.

[Learn more](#)

☐ Autoclass

Realiza la transición automáticamente de cada objeto a las clases Estándar o Nearline según la actividad a nivel de objeto para optimizar el costo y la latencia. Se recomienda si la frecuencia de uso puede ser impredecible. Se puede cambiar a una clase predeterminada en cualquier momento. [Detalles de precios](#)

☒ Configurar una clase predeterminada

Se aplica a todos los objetos del bucket, a menos que modifiques de forma manual la clase por objeto o establezcas reglas de ciclo de vida de los objetos. Se recomienda cuando el uso es muy predecible.

☒ Standard

La mejor opción para el almacenamiento a corto plazo y los datos de acceso frecuente

☐ Nearline

Ideal para copias de seguridad y datos a los que se accede menos de una vez al mes

☐ Coldline

Ideal para recuperación ante desastres y datos a los que se accede menos de una vez por trimestre

☐ Archive

La mejor opción para la conservación digital a largo plazo de los datos a los que se accede menos de una vez al año

Optimiza el almacenamiento para cargas de trabajo con uso intensivo de datos

☐ Habilita el espacio de nombres jerárquico en este bucket

Optimiza para IA o AA y estadísticas con una estructura jerárquica parecida a un sistema de archivos. Esta opción permanente permite cambiar los nombres a las carpetas atómicas, hacer fichas de carpetas de manera más rápida y otras mejoras no disponibles en los buckets estándar de Cloud Storage que usan un espacio de nombres plano.

**Figura 36. Selección del modo de almacenamiento de archivos.**

El acceso a la información se puede configurar como público o restringido mediante políticas del IAM.

### Elige cómo controlar el acceso a los objetos

#### Impedir el acceso público

Restringe el acceso público a los datos a través de Internet. Esto evitará que el bucket se use para el hosting web. [Más información](#)

☒ Aplicar la prevención de acceso público a este bucket

#### Control de acceso

☒ Uniforme

Garantiza el acceso uniforme a todos los objetos del bucket mediante el uso exclusivo de permisos a nivel de bucket (IAM). Esta opción se aplicará de manera permanente después de 90 días. [Más información](#)

☐ Preciso

Especifica el acceso a objetos individuales mediante el uso de permisos a nivel de objeto (LCA) además de los permisos a nivel de bucket (IAM). [Más información](#)

**Figura 37. Configuración de acceso Google Cloud.**

Además, se pueden establecer políticas de seguridad para proteger el contenido del bucket.

### Elige cómo proteger los datos de objeto

Tus datos siempre están protegidos con Cloud Storage, pero también puedes elegir entre estas opciones adicionales de protección de datos para agregar capas de seguridad extras.

#### Protección de datos

- ☒ **Política de eliminación no definitiva (para la recuperación de datos)**  
Cuando se habilita esta opción, el bucket y sus objetos se conservan durante un período específico después de que se borran, y pueden restablecerse durante este tiempo. [Más información](#)
- ☒ **Usar la duración de retención predeterminada**  
Todos los buckets tienen una duración de eliminación no definitiva de 7 días de forma predeterminada, a menos que el administrador de tu organización haya personalizado este valor.
- ☐ **Establecer duración de retención personalizada**  
Especifica por cuánto tiempo se deben conservar este bucket y sus objetos después de que se borren. Si estableces una duración de "0", se inhabilitará la eliminación no definitiva, lo que significa que se borrarán de forma permanente todos los objetos borrados.
- ☐ **Control de versiones de objetos (para el control de versión)**  
Para restablecer objetos borrados o reemplazados. Para minimizar el costo de almacenamiento de versiones, te recomendamos limitar la cantidad de versiones no actuales por objeto y programarlas para que venzan después de una cantidad de días. [Más información](#)
- ☐ **Retención (para el cumplimiento)**  
Para impedir que se borren o modifiquen los objetos del bucket durante un período específico.

*Figura 38. Políticas de seguridad Cloud Storage.*

## Microsoft Azure

**Microsoft Azure** es una plataforma de servicios en la nube proporcionada por Microsoft que ofrece un conjunto diverso de soluciones tecnológicas a través de internet. Azure es una de las principales plataformas en la nube utilizadas por empresas y desarrolladores para almacenar datos, ejecutar aplicaciones y gestionar infraestructuras, todo sin la necesidad de poseer hardware físico. Con una gran variedad de herramientas y servicios, Azure facilita desde la creación de aplicaciones hasta la gestión de redes y el análisis de datos.

Principales servicios que ofrece Azure:

### Cómputo:

**Azure Virtual Machines:** Permite crear y gestionar máquinas virtuales en la nube para ejecutar aplicaciones, con flexibilidad en el tamaño y los recursos asignados.

**Azure App Services:** Plataforma para el despliegue de aplicaciones web, APIs y backends sin necesidad de gestionar servidores.

**Azure Functions:** Servicio de computación sin servidor (serverless), permitiendo ejecutar fragmentos de código en respuesta a eventos sin la necesidad de gestionar infraestructura.

**Azure Kubernetes Service (AKS):** Servicio de administración de contenedores basado en Kubernetes, para gestionar aplicaciones en contenedores a gran escala.

## **Almacenamiento:**

**Azure Blob Storage:** Almacenamiento de objetos en la nube, ideal para almacenar grandes volúmenes de datos no estructurados.

**Azure Disk Storage:** Almacenamiento de discos persistentes para máquinas virtuales, usado principalmente en aplicaciones que requieren alta disponibilidad.

- **Azure Data Lake Storage:** Solución de almacenamiento de datos en bruto a gran escala para análisis.

## **Bases de datos:**

**Azure SQL Database:** Base de datos relacional completamente gestionada que soporta SQL Server y ofrece capacidades de escalabilidad y alta disponibilidad.

**Azure Cosmos DB:** Base de datos NoSQL completamente distribuida a nivel global con baja latencia y escalabilidad infinita.

**Azure Database for MySQL:** Servicio administrado para bases de datos MySQL en la nube.

## **Redes:**

**Azure Virtual Network (VNet):** Permite crear redes privadas en la nube y configurar la conectividad con otros servicios, dispositivos locales y entornos híbridos.

**Azure Load Balancer:** Distribuye el tráfico de red entrante entre los servidores virtuales de forma eficiente y escalable.

**Azure DNS:** Servicio que permite gestionar y resolver dominios de nombre para aplicaciones y servicios en Azure.

## **Desarrollo y DevOps:**

**Azure DevOps:** Herramientas para la gestión de proyectos, control de versiones, integración continua y despliegue continuo de aplicaciones.

**Azure Pipelines:** Servicio de integración y entrega continua (CI/CD) para automatizar los flujos de trabajo de desarrollo de software.

**Azure Logic Apps:** Herramienta para automatizar flujos de trabajo entre aplicaciones y servicios sin necesidad de escribir código.

## **Inteligencia Artificial y Aprendizaje Automático:**

**Azure Machine Learning:** Plataforma para crear, entrenar y desplegar modelos de aprendizaje automático con capacidades avanzadas.



**Azure Cognitive Services:** Conjunto de APIs preconstruidas que permiten incorporar funcionalidades de inteligencia artificial, como visión por computadora, análisis de texto y traducción, sin la necesidad de crear modelos desde cero.

### **Seguridad, Identidad y Cumplimiento:**

**Azure Active Directory (AAD):** Servicio de gestión de identidades y accesos para controlar quién tiene acceso a qué recursos dentro de la nube.

**Azure Security Center:** Herramienta para la supervisión de la seguridad de los recursos en Azure, que proporciona alertas sobre posibles vulnerabilidades y amenazas.

**Azure Key Vault:** Almacén seguro para la gestión de secretos, claves y certificados utilizados en las aplicaciones.

### **Monitoreo y Gestión:**

**Azure Monitor:** Solución integral para monitorear el rendimiento de las aplicaciones y recursos de Azure, proporcionando métricas, registros y alertas.

**Azure Log Analytics:** Servicio que permite analizar y consultar grandes volúmenes de datos generados por servicios de Azure y otras fuentes.

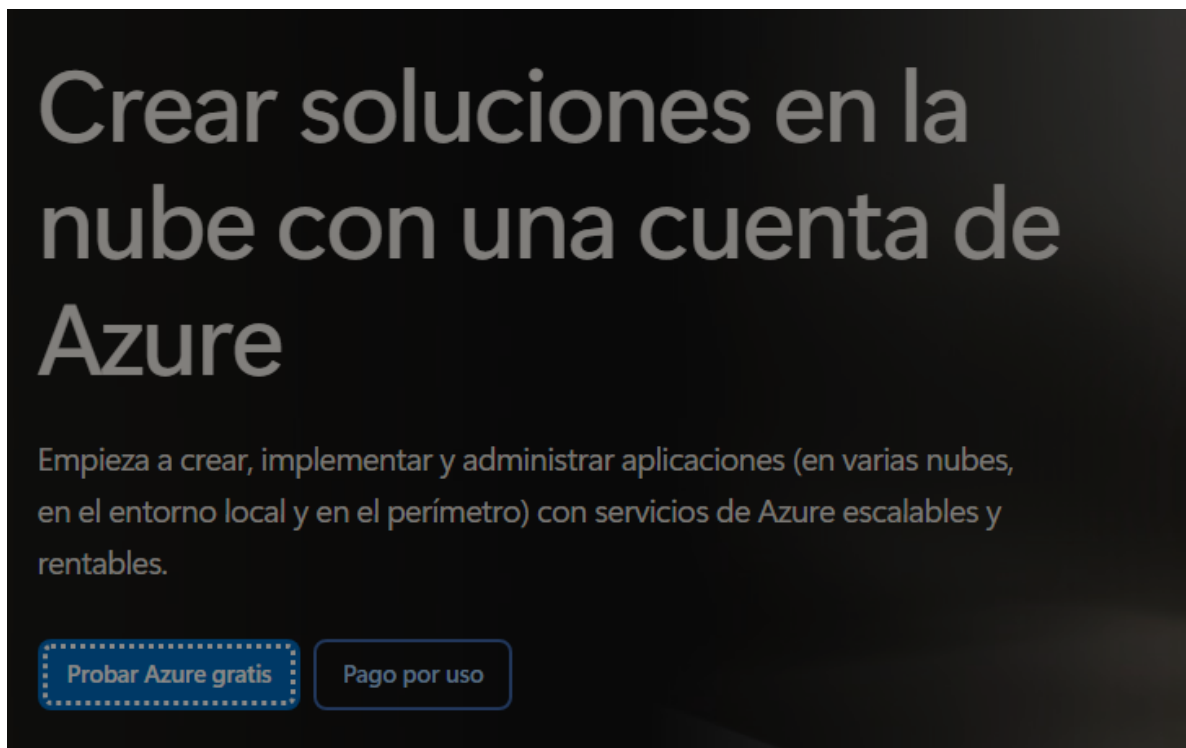
**Azure Automation:** Servicio para automatizar la gestión de tareas repetitivas, como la actualización de recursos y el control de configuraciones.

### *Configuración de una nube básica en azure*

#### *Creación de una cuenta de azure*

Para crear una cuenta en azure, podemos seguir los siguientes pasos:

1. **Acceder al portal de Azure:** Para comenzar con Microsoft Azure, hay que acceder al portal oficial: <https://portal.azure.com/>.
2. **Iniciar sesión o crear cuenta de Microsoft:** Si ya tenemos una cuenta de Microsoft (por ejemplo, una cuenta de Outlook o Xbox), podemos iniciar sesión directamente. Si no, necesitaremos crear una cuenta de Microsoft primero. Para ello, simplemente seleccionamos la opción de "**Crear una cuenta**" y seguimos las indicaciones para registrar nuestra información personal.
3. **Seleccionar "Probar Azure gratis":** Una vez que hayamos iniciado sesión, encontraremos la opción "**Probar Azure gratis**". Al hacer clic en esta opción, accederemos al formulario de registro para empezar a disfrutar de los beneficios gratuitos de Azure.

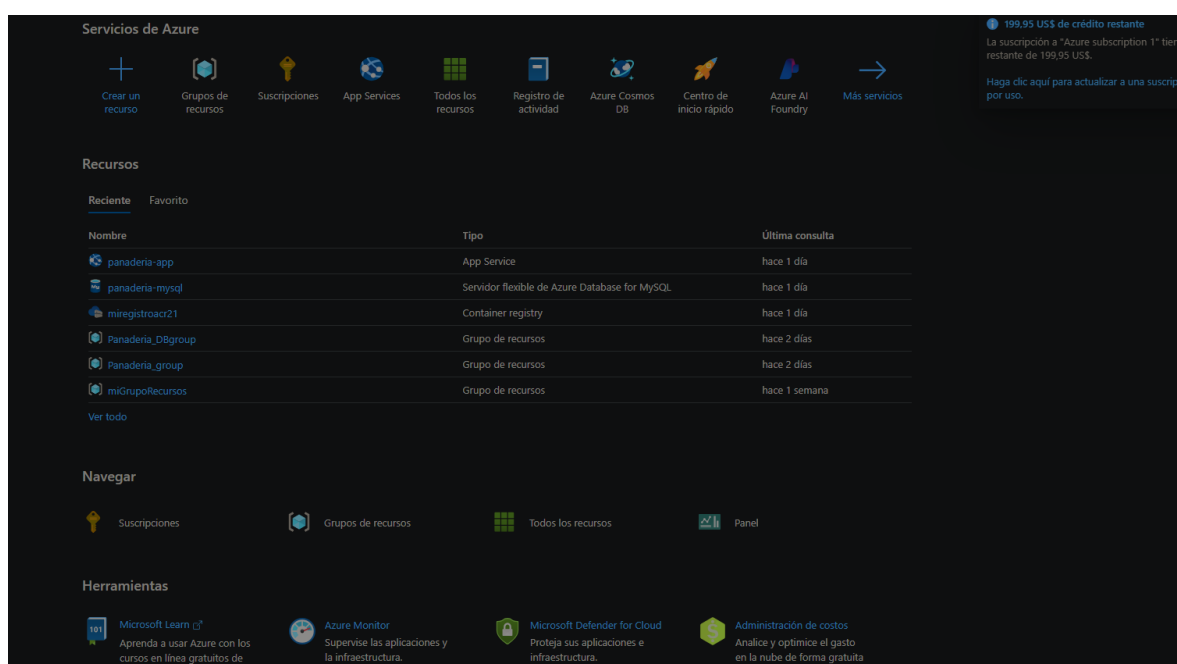


*Figura 39. Página de bienvenida de Microsoft Azure.*

4. **Proporcionar información personal y detalles de pago:** Se nos pedirá ingresar nuestros datos personales, como nombre, correo electrónico, dirección y un número de tarjeta de crédito o débito. Aunque es necesario proporcionar los datos de pago para verificar la cuenta, Azure no nos cobrará nada a menos que superemos los créditos gratuitos, y de igual manera en caso de hacerlo se nos avisara que podemos empezar a pagar para usar los servicios de azure.

*Figura 40. Pagina de registro de informacion del perfil.*

5. **Obtener los 200 USD en créditos gratuitos:** Al completar el proceso de registro, recibimos 200 USD en créditos para usar durante los primeros 30 días. Estos créditos nos permitirán explorar diversos servicios de Azure sin incurrir en costos adicionales, como crear máquinas virtuales, bases de datos, almacenamiento y más.
6. **Habilitar la autenticación multifactor (MFA):** Para mejorar la seguridad de la cuenta, es recomendable activar **la autenticación multifactor (MFA)**. Este paso ayuda a proteger nuestra cuenta de accesos no autorizados añadiendo una capa extra de seguridad.
7. **Explorar los servicios de Azure:** Una vez completado el registro y la configuración de seguridad, ya podemos comenzar a explorar todos los servicios que ofrece Azure. Podemos utilizar los 200 USD de crédito para crear y gestionar recursos como máquinas virtuales, bases de datos, servicios de almacenamiento, entre otros.



*Figura 41. Pagina de inicio de Azure, una vez creada una cuenta.*

### *¿Por qué se otorgan los 200 USD de crédito en Azure?*

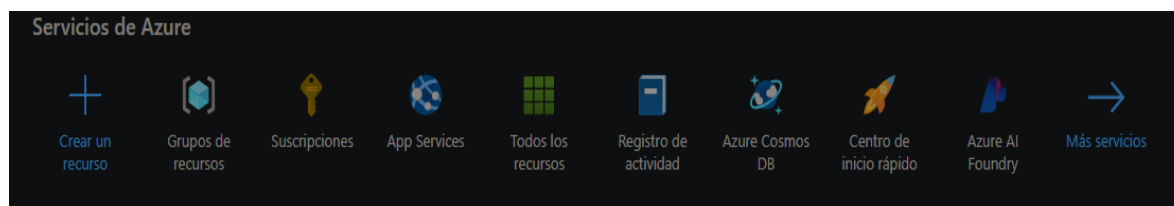
Los 200 USD de crédito gratuito sirven para que los nuevos usuarios puedan probar y familiarizarse con los servicios de Azure sin asumir riesgos financieros. Esta estrategia permite a los usuarios explorar las soluciones en la nube de Microsoft, experimentar con sus herramientas y, si lo desean, adoptar servicios de pago a largo plazo una vez que hayan agotado el crédito inicial o después de los 30 días.

De igual manera, una vez creada la cuenta en Azure, es recomendable seguir buenas prácticas de seguridad para proteger los recursos y la información. En lugar de utilizar la cuenta

principal para la administración diaria, se debe crear usuarios y grupos a través de **Azure Active Directory (AAD)**. Esto permite gestionar el acceso de manera más eficiente, asignando permisos específicos a cada usuario según sea necesario. Es importante aplicar el principio de **privilegio mínimo**, otorgando solo los permisos esenciales para que los usuarios realicen sus tareas, lo cual reduce los riesgos de accesos no autorizados.

Además, se debe organizar a los usuarios en grupos para gestionar los permisos de manera más eficiente. De esta manera, se facilita la administración y se asegura que los permisos se apliquen de manera uniforme. Por último, para mejorar la seguridad, es fundamental **habilitar la autenticación multifactor (MFA)**. Esta medida añade una capa extra de protección al requerir un segundo factor, como un código generado por una aplicación de autenticación o un mensaje SMS, lo que dificulta el acceso no autorizado, incluso si la contraseña se ve comprometida.

Para empezar, en la pantalla de inicio de Azure, si queremos ver todos los servicios ofrecidos por la plataforma, debemos hacer clic en "Más servicios". Esto nos llevará a una lista completa de los servicios disponibles, que incluye opciones para crear y gestionar recursos, monitoreo, análisis de datos, y mucho más. Ahora, con esta base, pasemos a la configuración de monitoreo de la nube.

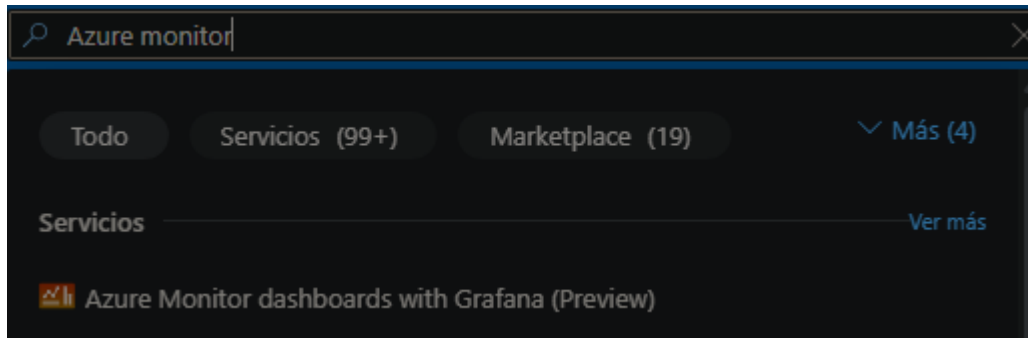


*Figura 42. Apartado donde se pueden ver los servicios de azure.*

## Configuración de Monitoreo de la Nube en Azure

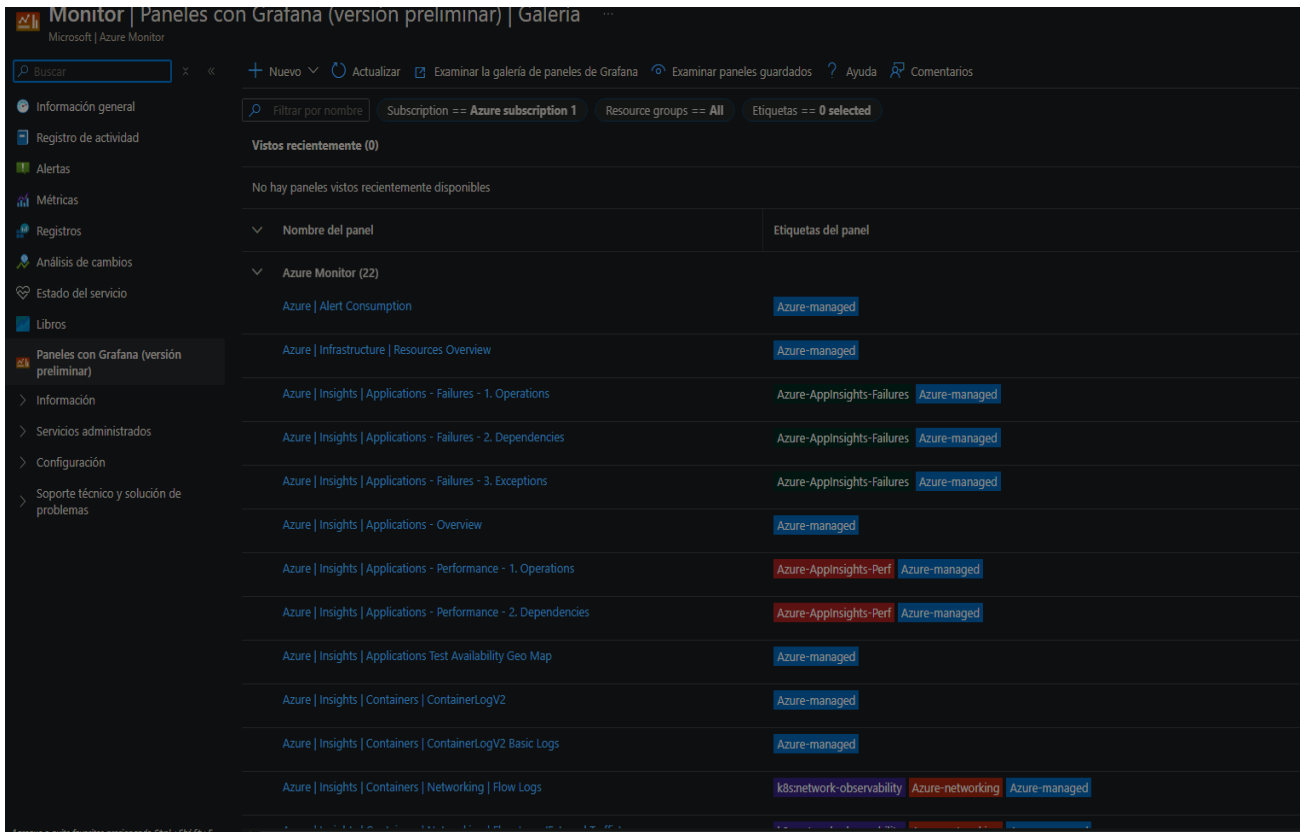
Azure ofrece herramientas integradas para monitorear la infraestructura y aplicaciones desplegadas, lo que permite obtener métricas de rendimiento, visualizar gráficas y configurar alertas de consumo. Uno de los principales servicios para esto es **Azure Monitor**.

1. **Acceder a Azure Monitor:** Para comenzar, desde el portal de Azure, podemos acceder a Azure Monitor a través del menú de servicios. Si ya estamos en el panel de inicio (como el que muestras en la imagen), podemos hacer clic en "Más servicios" y buscar "Monitor". También es posible acceder directamente a Azure Monitor desde la barra de búsqueda en la parte superior del portal.



*Figura 43. Búsqueda desde Azure del servicio.*

2. **Visualización de Gráficas y Métricas:** Dentro de Azure Monitor, encontraremos una variedad de métricas relacionadas con el uso de los recursos. Aquí podemos ver gráficas interactivas que muestran el desempeño de las máquinas virtuales, bases de datos, aplicaciones y otros recursos en la nube. Estas gráficas pueden incluir información sobre el uso de CPU, memoria, almacenamiento, tráfico de red, entre otros. Las métricas se actualizan en tiempo real y nos proporcionan una visión clara del rendimiento de nuestros recursos.



*Figura 44. Pagina principal de Azure monitor.*

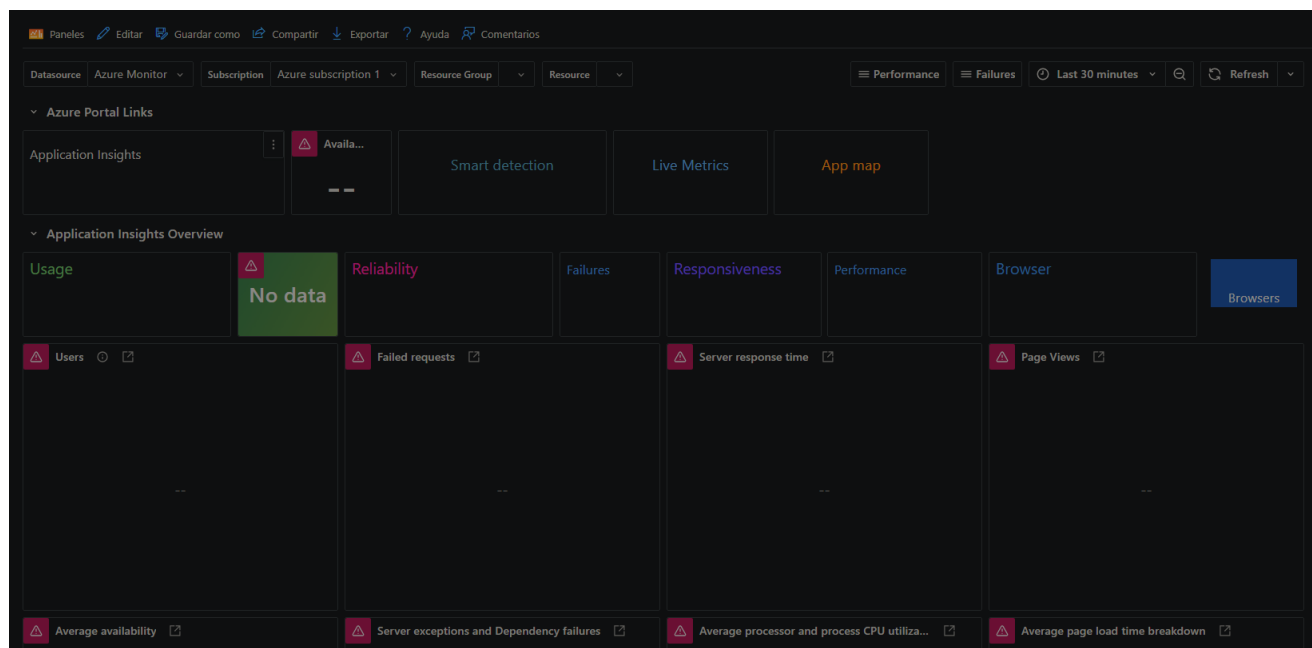
En la pantalla de Azure Monitor, podemos explorar los diferentes paneles preconfigurados que nos ofrece la plataforma, los cuales están organizados por categorías como Application

Insights, Containers, y Networking. Al buscar Azure Monitor, podemos ver una lista de paneles disponibles, como Azure Monitor, que nos da una visión general de los recursos, o Application Insights, donde podemos ver métricas de rendimiento de nuestras aplicaciones.

Por ejemplo, si seleccionamos uno de los paneles, como Azure Insights | Applications Overview, obtendremos un resumen completo sobre el estado de nuestras aplicaciones en la nube, incluyendo información de fallos y rendimiento. También podemos configurar estos paneles según nuestras necesidades, ajustando las métricas que queremos visualizar, lo que nos permite monitorear el rendimiento en tiempo real.

Además, podemos personalizar las alertas de consumo y los umbrales, de modo que si cualquier recurso supera el límite establecido, recibiremos notificaciones automáticas. Esto nos da un control total sobre el uso de los servicios de Azure, ayudándonos a gestionar los costos y la eficiencia operativa.

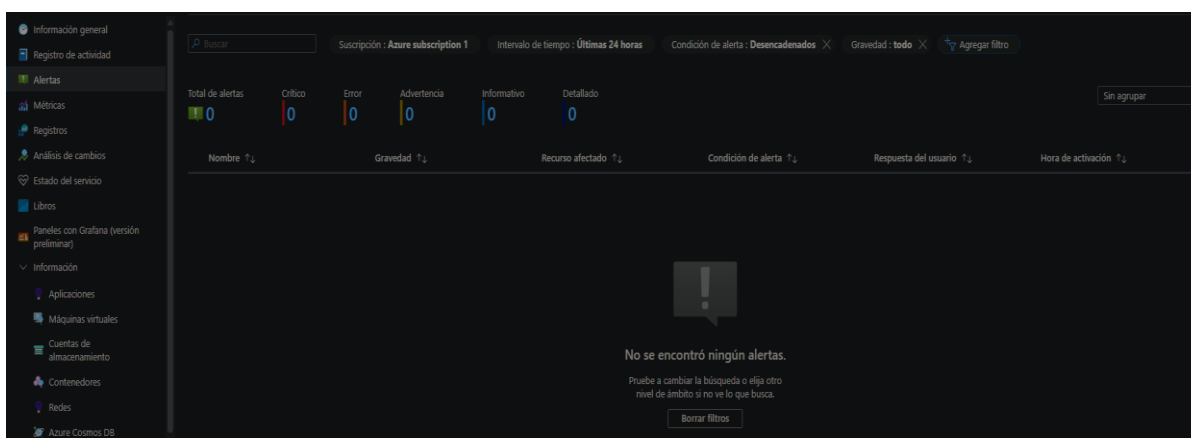
Cada panel de Azure Monitor está diseñado para mostrar información específica sobre diferentes métricas y servicios. Por ejemplo, al seleccionar el panel "Azure Insights | Applications Overview", podemos ver un resumen sobre el estado de nuestras aplicaciones, incluyendo el uso de recursos y la tasa de fallos. Este tipo de paneles es útil para obtener un panorama claro de cómo están funcionando nuestras aplicaciones.



*Figura 45. Panel de Azure Insights | Applications Overview.*

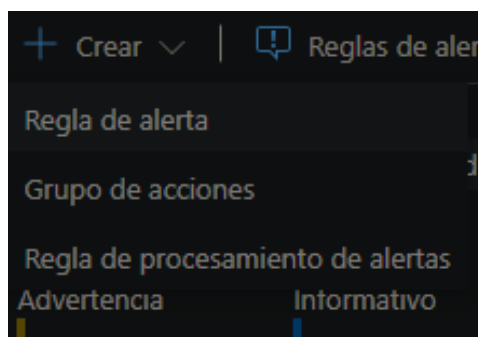
Además de la visualización de métricas, Azure Monitor también permite configurar alertas de consumo para que podamos recibir notificaciones cuando algún recurso exceda un umbral predefinido. Esto es útil para controlar el uso de recursos y evitar costos inesperados.

Por ejemplo, podemos configurar alertas para cuando el uso de CPU o almacenamiento de una máquina virtual alcance un determinado nivel, o si la base de datos comienza a superar su capacidad de uso.



*Figura 46. Panel de control de alertas en azure monitor.*

Dentro, podemos crear las distintas alertas como se ve a continuación:



*Figura 47. Creación de un tipo de alerta en azure monitor.*

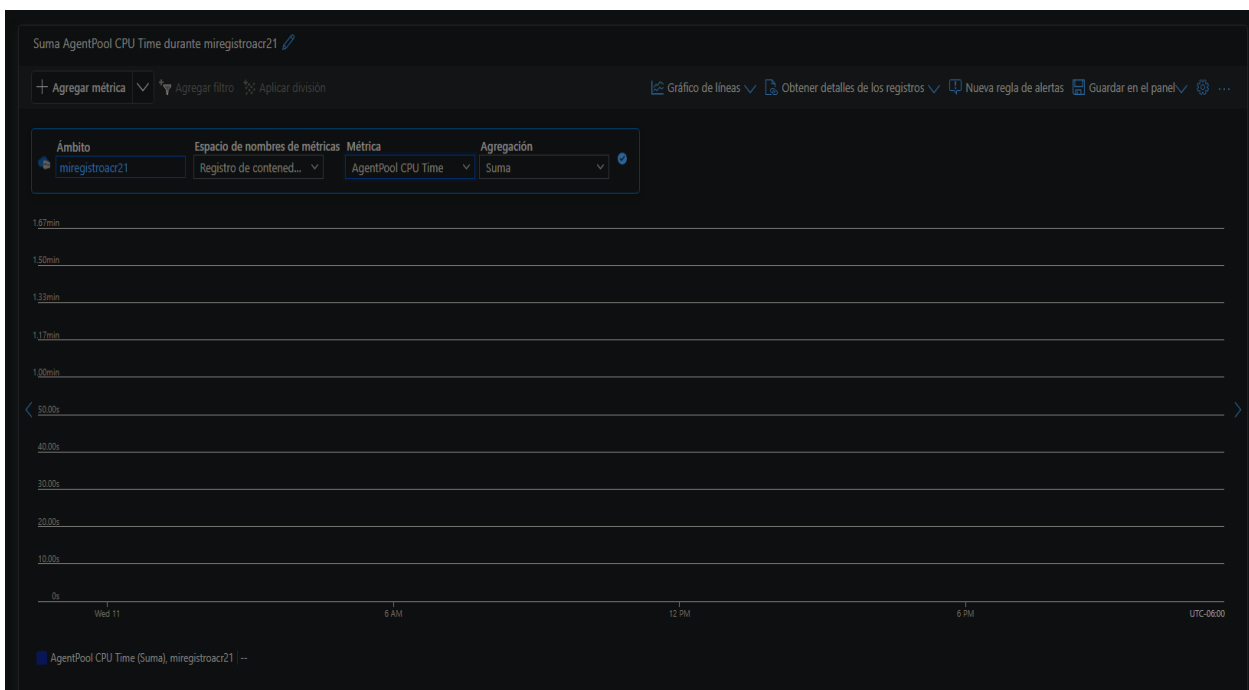
Dentro de Azure Monitor, podemos crear diferentes tipos de alertas para gestionar el estado y consumo de nuestros recursos. Al acceder a "Reglas de alerta" en el menú, se presentan varias opciones como "Regla de alerta", "Grupo de acciones" y "Regla de procesamiento de alertas". A continuación se explica qué se puede hacer con cada una de estas opciones:

1. **Regla de alerta:** Esta opción nos permite crear una nueva alerta personalizada. Al seleccionar "Regla de alerta", podemos definir el recurso a monitorear, establecer las condiciones que activarán la alerta (por ejemplo, si el uso de CPU supera cierto umbral) y elegir los canales de notificación como correo electrónico, SMS o mensajes

en Microsoft Teams. Es el paso fundamental para recibir alertas cuando se detecten cambios o eventos en los recursos.

2. **Grupo de acciones:** En esta sección, podemos organizar y gestionar acciones que se ejecutarán cuando se active una alerta. Un grupo de acciones puede incluir actividades como enviar un correo de notificación, ejecutar un script o realizar otras tareas automatizadas que ayuden a mitigar el problema. Aquí se definen los contactos o sistemas a los que se les enviarán las notificaciones.
3. **Regla de procesamiento de alertas:** Esta opción nos permite configurar cómo se procesarán las alertas una vez generadas. Podemos establecer reglas adicionales, como la frecuencia con la que se envían las alertas o definir umbrales más complejos que se deben cumplir antes de generar una alerta. También permite crear políticas para manejar situaciones de manera más eficiente y reducir la cantidad de alertas redundantes.

De igual manera, los paneles de Azure Monitor también permiten la visualización interactiva de gráficas en tiempo real, lo que facilita el análisis de las métricas de nuestros recursos. Estas gráficas pueden ser personalizadas para mostrar información sobre el uso de CPU, memoria, tráfico de red, entre otros.



*Figura 48. Gráficas de Métricas en Azure Monitor.*

En general al utilizar, Azure Monitor, podemos explorar diferentes paneles para visualizar el estado y rendimiento de nuestras aplicaciones y recursos. Además, podemos configurar alertas de consumo para mantenernos informados sobre el uso de recursos y evitar costos



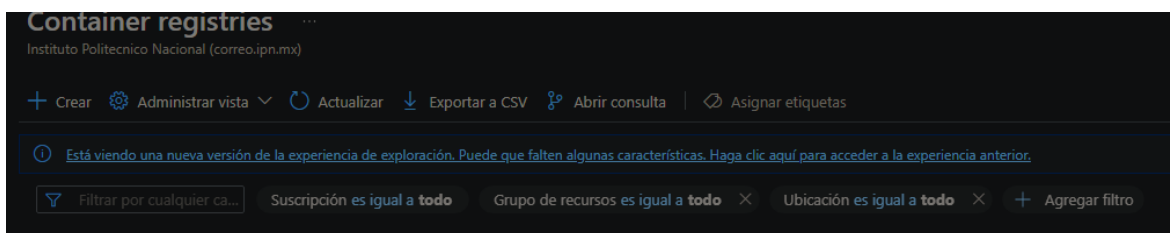
innecesarios. Las gráficas interactivas también nos permiten analizar los datos en tiempo real, proporcionando una visión clara y precisa del estado de nuestra infraestructura en la nube.

## Creación de un Container Registry en Azure

Para crear un Azure Container Registry, lo primero que debemos hacer es buscar "Container Registry" en el portal de Azure. Esto nos llevará a la opción de crear un nuevo Container Registry.

### 1. Acceder al Container Registry:

- En el portal de Azure, busca "Container Registry" en la barra de búsqueda.
- Selecciona la opción de "Container Registries" y haz clic en "Crear" para iniciar el proceso de creación.



*Figura 49. Pagina para la creación de un container registries.*

### 2. Información requerida al crear el Container Registry:

- **Nombre:** Es importante proporcionar un nombre único para el registro de contenedores. Este nombre será utilizado para acceder al **Container Registry**.
- **Grupo de recursos:** Debemos seleccionar un **grupo de recursos** existente o crear uno nuevo para asociar el registro de contenedores.
- **Región:** Es necesario seleccionar una región. Es recomendable que esta región coincida con la de otros recursos que vayamos a crear, como una **App Service** o bases de datos, para optimizar el rendimiento y evitar problemas de latencia.
- **Plan de precios:** Debemos seleccionar el **plan de precios** que mejor se adapte a nuestras necesidades. Azure ofrece diferentes opciones de precios según el tamaño y la frecuencia de uso.
- **Personalización de redes y cifrado:** La personalización de redes y la habilitación de cifrado están disponibles en planes premium. Si no seleccionamos un plan premium, podemos omitir estas opciones.

- **Etiquetas:** Podemos agregar etiquetas para organizar el recurso y facilitar la gestión.

Para crear el Container Registry de manera básica, simplemente es necesario proporcionar el nombre del registro, seleccionar el grupo de recursos y la región, y elegir el plan de precios más adecuado.

**Crear Registro de contenedor**

**Datos básicos** | Redes | Cifrado | Etiquetas | Revisar y crear

Azure Container Registry permite compilar, almacenar y administrar artefactos e imágenes de contenedor en un registro privado para todos los tipos de implementación de contenedor. Use registros de contenedor de Azure con sus canalizaciones de desarrollo e implementación de contenedores actuales. Use Azure Container Registry Tasks para compilar imágenes de contenedor en Azure a petición, o bien automatizar compilaciones desencadenadas por actualizaciones del código fuente, actualizaciones de la imagen base de un contenedor o temporizadores. [Más información](#)

**Detalles del proyecto**

Suscripción \* Azure subscription 1

Grupo de recursos \* [Crear nuevo](#)

**Detalles de la instancia**

Nombre del Registro \* Escriba el nombre .azurecr.io

Ubicación \* Canada Central

Ámbito de etiqueta de nombre de dominio \* ① No seguro

Nombre de dominio del Registro

Uso de zonas de disponibilidad ① ☐   
 Las zonas de disponibilidad se activan en registros premium y en regiones que admiten zonas de disponibilidad. [Más información](#)

Plan de precios \* ① Estándar

Modo de permisos de asignación de roles (vista previa) ①   
☐ Permisos del registro RBAC + repositorio ABAC   
☒ Permisos del Registro RBAC

[Revisar y crear](#) < Anterior Siguiente: Redes >

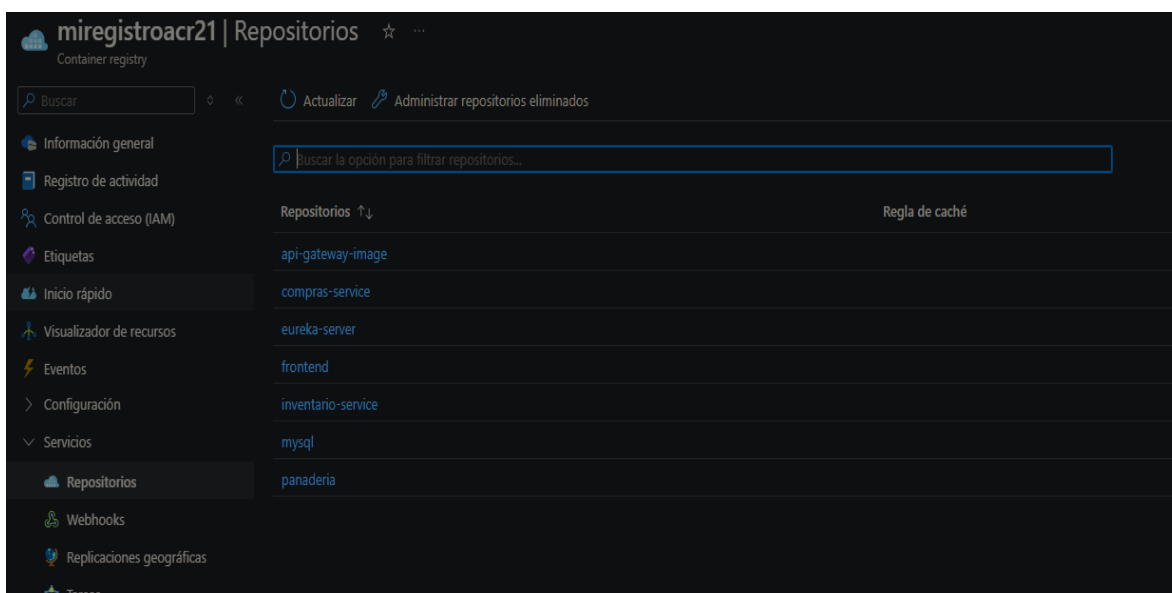
*Figura 50. Detalles del container registry.*

### 3. Crear el Container Registry:

- Después de completar la configuración, revisamos la información y hacemos clic en "**Revisar + Crear**".
- Finalmente, seleccionamos "**Crear**" para desplegar el **Container Registry**.

El Container Registry que hemos creado se utilizará para almacenar y gestionar imágenes de contenedores. Esto es útil si estamos trabajando con aplicaciones basadas en contenedores, como Docker, y queremos gestionar de manera centralizada nuestras imágenes para facilitar el despliegue.

Dentro de dicho container registry, podemos empezar a subir las imágenes de Docker. Esto se realiza mediante la autenticación de nuestra terminal con el Container Registry, y luego etiquetando y subiendo nuestras imágenes usando Docker CLI. Al hacerlo, las imágenes quedarán almacenadas en el Azure Container Registry, listas para ser desplegadas en otros servicios de Azure, como App Service o Azure Kubernetes Service (AKS) y se vería en Azure de la siguiente manera, donde cada imagen estaría correctamente etiquetada en la nube.



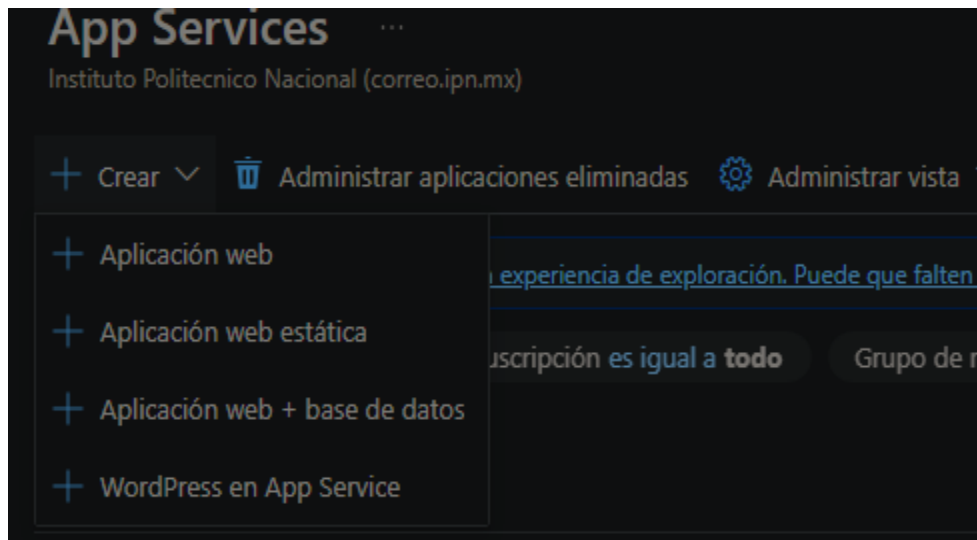
*Figura 51. Imágenes subidas a la nube en el container registry.*

## Creación de un App Service en Azure

El App Service de Azure es una plataforma que permite crear, implementar y administrar aplicaciones web. En este caso, utilizaremos el App Service para ejecutar contenedores Docker, pero también es útil para una variedad de otros escenarios.

### 1. Acceder a App Service:

- En el portal de **Azure**, busca "**App Service**" y selecciona la opción de "**Crear**" para iniciar la creación de un nuevo **App Service**.



*Figura 52. Creación de un App Services en Azure.*

2. **Tipos de aplicaciones en App Service:** Al momento de crear un App Service, se nos presenta la opción de elegir diferentes tipos de aplicaciones:
  - **Aplicación web:** Para desplegar una aplicación web tradicional.
  - **Aplicación web estática:** Para sitios web estáticos, sin necesidad de backend dinámico.
  - **Aplicación web + base de datos:** Permite combinar la aplicación con una base de datos como parte de la creación.
  - **WordPress en App Service:** Si seleccionamos esta opción, se integrará con una base de datos MySQL para crear un sitio de WordPress.

El tipo más común de App Service que se utiliza es el de "Aplicación web", que es ideal para desplegar contenedores Docker o aplicaciones web tradicionales.

3. **Información solicitada para crear el App Service:** Durante el proceso de creación, se nos solicita la siguiente información:
  - **Grupo de recursos:** Seleccionamos el grupo de recursos que queremos asociar al **App Service**.
  - **Nombre:** Proporcionamos un nombre único para nuestra aplicación web.
  - **Código o contenedor:** Elegimos si queremos desplegar una aplicación basada en **código** o en un **contenedor Docker**.
  - **Sistema operativo:** Si elegimos contenedor Docker, el sistema operativo si o si será Linux, ya que Docker generalmente funciona en Linux.

- **Región:** Seleccionamos la misma región donde hemos creado otros recursos, como el Container Registry y la base de datos, para asegurar que todos los servicios estén cerca y optimicen el rendimiento.
- **Plan de precios:** Seleccionamos el plan gratuito si no necesitamos capacidades avanzadas.

**Crear aplicación web**

[Datos básicos](#) | [Base de datos](#) | [Implementación](#) | [Redes](#) | [Supervisión y protección](#) | [Etiquetas](#) | [Revisar y crear](#)

App Service Web Apps le permite generar, implementar y escalar rápidamente aplicaciones empresariales web, móviles y de API que se ejecutan en cualquier plataforma. Satisface los estrictos requisitos de rendimiento, escalabilidad, seguridad y cumplimiento sin renunciar a una plataforma totalmente administrada para el mantenimiento de la infraestructura. [Más información](#)

**Detalles del proyecto**

Seleccione una suscripción para administrar los recursos implementados y los costos. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción \* ⓘ Azure subscription 1 ▼

Grupo de recursos \* ⓘ (Nuevo) Grupo de recursos ▼

[Crear nuevo](#)

**Detalles de instancia**

Nombre Nombre de la aplicación web .azurewebsites.net

☒ Nombre de host predeterminado único seguro activado. [Más información sobre esta actualización](#)

Publicar \* ☒ Código ☐ Contenedor

Pila del entorno en tiempo de ejecución \* Seleccione una pila del entorno en tiempo de ejecución ▼

Sistema operativo ☒ Linux ☐ Windows

Región Cargando... ⓘ

**Planes de precios**

El plan de tarifa de App Service determina la ubicación, las características, los costos y los recursos del proceso asociados a la aplicación. [Más información](#)

Plan de Linux ⓘ Cargando... ⓘ

[Revisar y crear](#) [< Anterior](#) [Siguiente: Base de datos >](#)

*Figura 53. Detalles del app services.*

#### 4. Otras opciones que permite configurar:

- **Base de datos:** Si lo deseamos, podemos integrar un sistema de base de datos con el App Service, como MySQL o Azure SQL Database.
- **Implementación con GitHub Actions:** Si queremos automatizar la implementación con GitHub Actions, debemos seleccionar otro tipo de pila de entorno en tiempo de ejecución.

#### 5. Redes:

- El App Service puede tener una dirección pública a Internet o estar aislado en una red virtual de Azure. También podemos configurar si la aplicación debe ser accesible públicamente o si debe estar protegida.

#### 6. Supervisión y protección:

- **Application Insights:** Nos permite habilitar la supervisión de la aplicación para detectar problemas de rendimiento y analizar el comportamiento de los usuarios.
- **Microsoft Defender for Cloud:** Agrega una capa de protección adicional al **App Service**, supervisando y protegiendo contra amenazas de seguridad.

#### 7. Etiquetas: Si queremos, podemos agregar etiquetas para organizar el recurso.

#### 8. Revisar y crear:

- Después de configurar todos los parámetros, revisamos la información y seleccionamos "**Crear**" para desplegar el **App Service**.

Cabe mencionar que, en caso de seleccionar contenedores (Docker), hay un apartado en el App Service donde se nos pide seleccionar el origen preferido para las imágenes de contenedor. Si ya hemos subido nuestras imágenes previamente al Azure Container Registry, podemos seleccionarlás directamente desde el registro al configurar el App Service. Si elegimos Azure Container Registry como origen, debemos autenticarlo mediante Managed Identity o Admin credentials. En caso de usar Managed Identity, los campos de imagen y etiqueta no se completarán automáticamente, por lo que tendremos que ingresar estos valores manualmente. Además, podemos proporcionar un comando de inicio personalizado para el contenedor, y si es necesario, habilitar la compatibilidad con sidecar para una configuración avanzada.

The screenshot shows the 'Contenedor' tab in the Azure portal. At the top, there are tabs for 'Datos básicos', 'Base de datos', 'Contenedor', 'Redes', 'Supervisión y protección', 'Etiquetas', and 'Revisar y crear'. Below the tabs, a message states: 'Seleccione el origen preferido para las imágenes de contenedor. Puede cambiar esta configuración y otras dependencias después de crear la aplicación. [Más información](#)'. The 'Compatibilidad con Sidecar' section has a toggle switch set to 'Off', with text indicating 'Configuración mejorada con compatibilidad con sidecar desactivada' and a link for 'Más información'. The 'Origen de imagen' section has three radio buttons: 'Inicio rápido' (unselected), 'Azure Container Registry' (selected), and 'Otros registros de contenedor' (unselected). The 'Opciones de Azure Container Registry' section includes a 'Registro' dropdown menu set to 'miregistroacr21', an 'Authentication' section with 'Managed identity' selected and 'Admin credentials' unselected, and an 'Identity' dropdown menu set to 'Cargando...'. A blue information box contains the text: 'When using managed identity, image and tag fields will not auto populate. Please manually enter the image and tag below.' Below this, there are three input fields: 'Imagen' (empty), 'Etiqueta' (empty), and 'Comando de inicio' (containing the example command: '/bin/bash; -c; echo hello; sleep 10000').

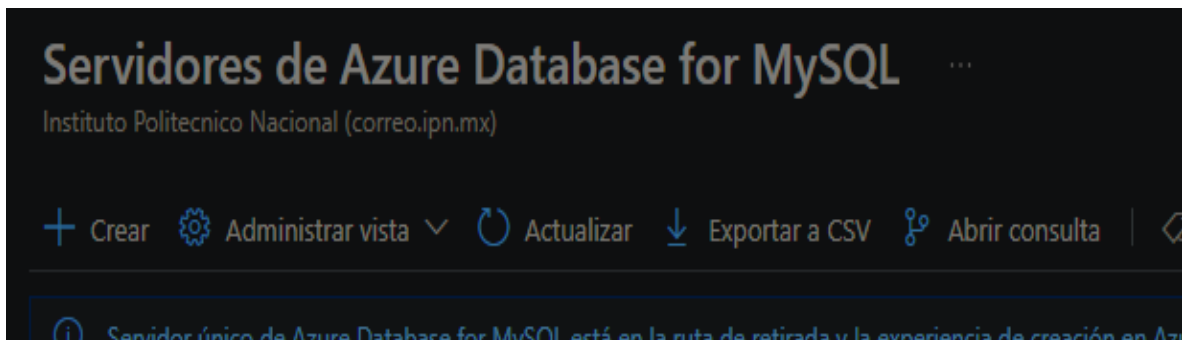
*Figura 54. Configuración de las imágenes del contenedor.*

## Creación de una Base de Datos con Azure Database for MySQL

Para crear una base de datos MySQL en Azure, podemos utilizar Azure Database for MySQL. Esto nos permitirá tener una base de datos gestionada en la nube para nuestras aplicaciones.

### 1. Acceder a Azure Database for MySQL:

- En el portal de Azure, buscamos "Azure Database for MySQL" y seleccionamos "Crear".



*Figura 55. Creación de un servidor de BD MySQL en Azure.*

## 2. Seleccionar cómo usar el servicio:

- Al darle crear, se nos pregunta cómo pensamos usar el servicio:
  - **Servidor flexible:** Ideal para cargas de trabajo de producción que requieren alta disponibilidad, rendimiento predecible y optimización de costos.
  - **WordPress + servidor flexible de MySQL:** Si seleccionamos esta opción, se instalará WordPress en App Service y una base de datos MySQL gestionada en Azure.

## 3. Configuración del servidor flexible:

- **Grupo de recursos:** Seleccionamos el grupo de recursos donde se almacenará el servidor.
- **Nombre:** Proporcionamos un nombre único para la base de datos.
- **Región:** Seleccionamos la misma región que los otros recursos (como el **App Service** y el **Container Registry**).
- **Credenciales:** Ingresamos las credenciales de acceso a la base de datos (usuario y contraseña). Cabe mencionar que no permite credenciales básicas como root, admin entre otras. Tiene que contener el usuario un mínimo de caracteres con letras y números y la contraseña un mínimo de 8 caracteres entre mayúsculas, minúsculas, números y símbolos especiales.



Servidor flexible

Microsoft

Datos básicos

Etiquetas

Revisar

Cree un servidor de Azure Database for MySQL.

Detalles del proyecto

Seleccione la suscripción para administrar recursos implementados y los costes. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción \* ⓘ

Azure subscription 1

Grupo de recursos \* ⓘ

DefaultResourceGroup-CCAN

Crear nuevo

Detalles del servidor

Especifique la configuración necesaria para este servidor, incluida la selección de una ubicación y la configuración de los recursos de proceso y almacenamiento.

Nombre del servidor \* ⓘ

Escribir nombre del servidor

Región \* ⓘ

(Canada) Canada Central

Zona de disponibilidad \* ⓘ

Sin preferencias

Autenticación

Inicio de sesión del administrador \* ⓘ

Escriba el inicio de sesión del administrador

Contraseña \* ⓘ

Escribir contraseña

Confirmar contraseña \*

Confirmar la contraseña

Detalles de la carga de trabajo (Comparar tipos de carga de trabajo)

Tipo de carga de trabajo \* ⓘ

Desarrollo/pruebas

Estándar

Enterprise

Elija uno de estos tipos de carga de trabajo para configurar rápidamente el servidor según sus necesidades. Puede modificar la configuración después de crearlo.

☐

Aprender regla de firewall para la dirección IP actual

Revisar y crear

Siguientes Etiquetas >

Estimated costs

SKU de proceso

USD -/mes

Standard\_B1ms

-

Almacenamiento

USD -/mes

Almacenamiento seleccionado - GiB (USD - por GiB)

-

IOPS de escalado automático

Las IOPS de escalado automático se facturan según el uso por millón de incrementos de solicitud. [Más información](#)

Retención de la copia de seguridad

La retención de copias de seguridad se factura en función del almacenamiento adicional usado para conservar las copias de seguridad. [Más información](#)

Bandwidth

En el caso de la transferencia de datos saliente entre servicios de distintas regiones, se aplicarán cargos adicionales. Cualquier transferencia de datos entrante es gratuita. [Más información](#)

Total estimado

-

Los precios solo reflejan una estimación. [Vea la calculadora de precios de Azure](#). Los cargos finales aparecerán en la moneda local en las vistas de facturación y análisis de costos.

Figura 56. Configuración de la BD MySQL en Azure.

#### 4. Alta disponibilidad:

- Alta disponibilidad: Podemos habilitar alta disponibilidad para garantizar que la base de datos sea resistente a fallos.
- Reglas de firewall: Añadimos reglas de firewall para permitir el acceso desde nuestras IPs específicas o desde Azure. Esto también se puede modificar posteriormente y añadir más reglas de firewall.

57

☐ Agregar regla de firewall para la dirección IP actual

**Alta disponibilidad**

La alta disponibilidad de "en la misma zona" y de "con redundancia de zona" brindan resistencia adicional al servidor en caso de que ocurra un error.

Habilitar alta disponibilidad \* ⓘ ☐

**Configuración de copia de seguridad**

Opción de redundancia de copia de seguridad ⓘ Redundancia local

Redundancia geográfica \* ⓘ ☐ Recuperarse de una interrupción o desastre regional

¿Desea obtener más información sobre su carga de trabajo de MySQL? (Opcional) ⓘ

*Figura 57. Configuración de la BD MySQL en Azure.*

## 5. Copia de seguridad:

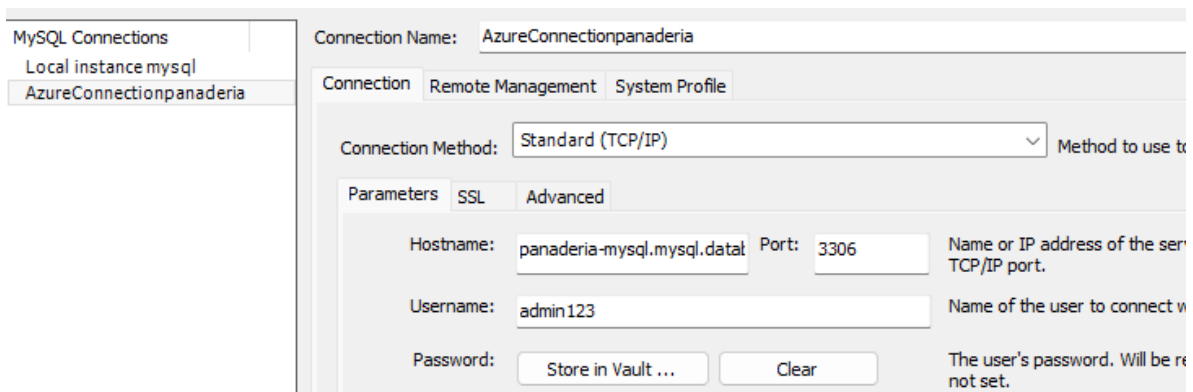
- Seleccionamos las opciones de redundancia de copia de seguridad: **Redundancia local** o **Redundancia geográfica**.

## 6. Etiquetas: Podemos añadir etiquetas para gestionar los recursos de manera más eficiente.

## 7. Revisar y crear:

- Después de completar la configuración, revisamos todo y seleccionamos **"Crear"** para desplegar el servicio de base de datos **MySQL**.

Una vez que hemos creado nuestra base de datos en Azure Database for MySQL, una manera de acceder a ella es utilizando MySQL Workbench. Para hacerlo, debemos crear una nueva conexión en MySQL Workbench con la URL de nuestra base de datos. Primero, obtenemos el Endpoint de la base de datos desde el portal de Azure, que generalmente tendrá el formato <nombre\_base\_de\_datos>.mysql.database.azure.com. Luego, en MySQL Workbench, seleccionamos "Nueva conexión", ingresamos el Endpoint como host, proporcionamos el nombre de usuario y la contraseña que configuramos al crear la base de datos, y finalmente, hacemos clic en "Test Connection" para verificar que la conexión se haya establecido correctamente. Si la prueba es exitosa, podemos comenzar a gestionar y ejecutar consultas en nuestra base de datos directamente desde MySQL Workbench.



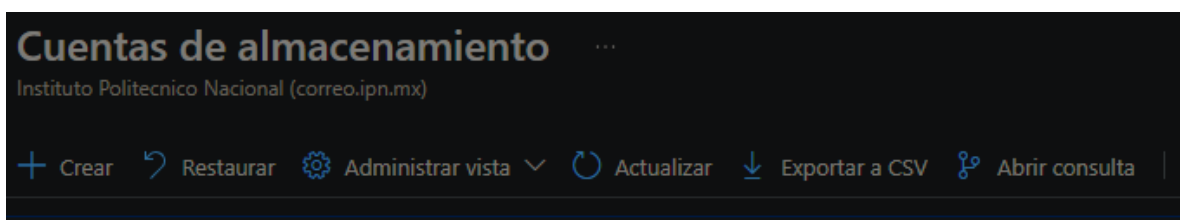
*Figura 58. Conexión a la BD desde MySQL Workbench.*

## Creación de una Cuenta de Almacenamiento en Azure

Una vez en el portal de Azure, si necesitamos crear un recurso de almacenamiento para guardar hasta 500 TB de datos en la nube, podemos optar por crear una cuenta de almacenamiento. Esta cuenta nos permitirá utilizar diversas opciones para almacenar diferentes tipos de datos, ya sea de objetos, archivos o NoSQL.

### 1. Acceder a Cuentas de Almacenamiento:

- En el portal de **Azure**, buscamos "**Cuentas de almacenamiento**" en la barra de búsqueda y seleccionamos "**Cuentas de almacenamiento**".
- Hacemos clic en "**Crear**" para comenzar el proceso de creación de la cuenta de almacenamiento.



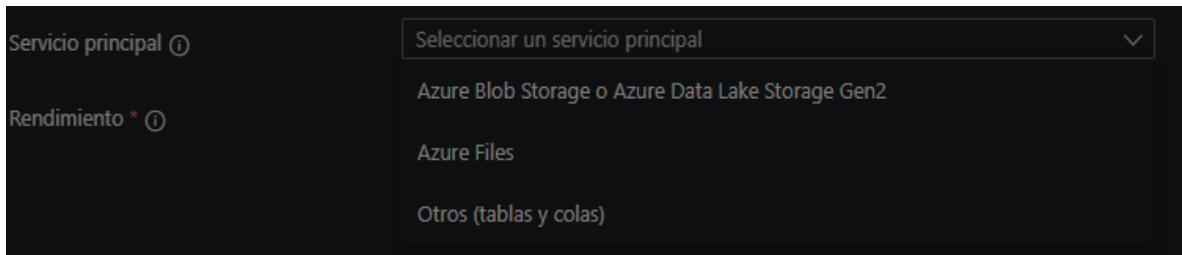
*Figura 59. Creación de una cuenta de almacenamiento.*

### 2. Tipos de cuentas de almacenamiento:

Al crear una cuenta de almacenamiento, tenemos varias opciones dependiendo del uso que le queramos dar:

- **Cuenta de Almacenamiento de Uso General:** Ideal para guardar **datos de objetos**, **almacenamiento NoSQL**, y configurar **colas de mensajes** para procesar trabajos asíncronos. Esta opción es flexible y permite trabajar con una variedad de tipos de datos.
- **Blob Storage:** Esta opción es más adecuada si estamos principalmente interesados en almacenar grandes cantidades de **datos no estructurados**, como imágenes, videos, y backups. Aquí podemos optimizar los costos

seleccionando entre los niveles de acceso **frecuente** o **esporádico** dependiendo de la frecuencia con la que accedemos a los datos.



*Figura 60. Tipos de cuenta de almacenamiento.*

3. **Configuración de la cuenta de almacenamiento:** Para crear la cuenta de almacenamiento, se nos solicita ingresar la siguiente información:

- **Nombre:** Proporcionamos un nombre único para la cuenta de almacenamiento, que será utilizado para acceder a los datos almacenados.
- **Grupo de recursos:** Seleccionamos un **grupo de recursos** donde se almacenará la cuenta.
- **Región:** Seleccionamos la **región** donde se ubicará la cuenta de almacenamiento. Es recomendable elegir la misma región que otros recursos que vayamos a utilizar, como bases de datos o aplicaciones.
- **Plan de precios:** Elegimos el plan de precios que mejor se adapte a nuestras necesidades. Esto dependerá de la cantidad de datos que planeamos almacenar y el nivel de acceso que requerimos (frecuente o esporádico).
- **Tipo de almacenamiento:** Aquí seleccionamos si queremos una cuenta de uso general o si vamos a utilizar Blob Storage.

4. **Características avanzadas:**

## Seguridad

- **Requerir transferencia segura para las operaciones de API de REST:** Esta opción permite que todas las operaciones de API se realicen de manera segura, utilizando HTTPS en lugar de HTTP.
- **Permitir el acceso anónimo en contenedores individuales:** Se puede habilitar o deshabilitar el acceso anónimo a contenedores específicos, lo cual es útil para controlar el acceso a los datos almacenados.

- **Habilitar el acceso a la clave de la cuenta de almacenamiento:** Permite acceder a las claves de la cuenta de almacenamiento para su uso en la autenticación de acceso a los recursos.
- **Versión mínima de TLS:** Esta configuración permite definir la versión mínima de TLS que se utilizará para las conexiones con la cuenta de almacenamiento.
- **Espacio de nombres jerárquico:** Al habilitar esta opción, se habilita la semántica de archivos y directorios, lo que es útil para cargas de trabajo de análisis de macrodatos, además de habilitar las listas de control de acceso (ACL).
- **Habilitar SFTP:** Si se habilita, se puede acceder a la cuenta de almacenamiento usando el protocolo SFTP, pero esta opción solo está disponible si el espacio de nombres jerárquico está habilitado.
- **Habilitar el sistema de archivos de red v3 (NFS v3):** Permite usar NFS v3 para acceder a los recursos de almacenamiento, pero también requiere que se habilite el espacio de nombres jerárquico.

Datos básicos   **Avanzado**   Redes   Protección de datos   Cifrado   Etiquetas   Revisar y crear

**Seguridad**

Permite establecer opciones de configuración de seguridad que afectan a su cuenta de almacenamiento.

Requerir transferencia segura para las operaciones de API de REST ☒ ⓘ

Permitir el acceso anónimo en contenedores individuales ☐ ⓘ

Habilitar el acceso a la clave de la cuenta de almacenamiento ☒ ⓘ

El valor predeterminado es la autorización de Microsoft Entra en Azure Portal ☐ ⓘ

Versión mínima de TLS ⓘ   Versión 1.2 ▼

Ámbito permitido para las operaciones de copia (versión preliminar) ⓘ   Desde cualquier cuenta de almacenamiento ▼

**Espacio de nombres jerárquico**

El espacio de nombres jerárquico, complementado con el punto de conexión de Data Lake Storage Gen2, habilita la semántica de archivos y directorios, acelera las cargas de trabajo de análisis de macrodatos y habilita las listas de control de acceso (ACL) [Obtener más información](#) ⓘ

Habilitar el espacio de nombres jerárquico ☐ ⓘ

**Protocolos de acceso**

Los puntos de conexión de Blob y Data Lake Gen2 se aprovisionan de forma predeterminada. [Obtener más información](#) ⓘ

Habilitar SFTP ⓘ   ⓘ SFTP solo se puede habilitar para cuentas de espacio de nombres jerárquicos

Habilitar el sistema de archivos de red v3 ⓘ   ⓘ Para habilitar NFS v3, se debe habilitar el "espacio de nombres jerárquico". [Más](#)

*Figura 61. Configuración de seguridad.*

## Redes

- **Conectividad de red:** Permite definir si la cuenta de almacenamiento será accesible de manera pública (con puntos de conexión de servicio o direcciones IP públicas) o de forma privada mediante un **punto de conexión privado**.
- **Punto de conexión privado:** Al habilitar esta opción, se crea un punto de conexión privado que permite acceder a la cuenta de almacenamiento de manera privada a través de **Private Link**.
- **Enrutamiento de red:** Permite seleccionar cómo se debe redirigir el tráfico desde el origen al punto de conexión de Azure, con opciones de enrutamiento de Microsoft recomendadas para la mayoría de clientes.

The screenshot shows the 'Redes' (Network) tab in the Azure portal for a storage account. The 'Conectividad de red' (Network connectivity) section is active, showing three radio button options for 'Acceso a la red' (Network access): 'Habilitar el acceso público desde todas las redes' (selected), 'Habilitar el acceso público desde redes virtuales y direcciones IP seleccionadas', and 'Deshabilitar el acceso público y usar el acceso privado'. Below this is the 'Punto de conexión privado' (Private endpoint) section, which includes a '+ Agregar punto de conexión privado' button and a table with columns: Nombre, Suscripción, Grupo de r..., Región, Tipo de su..., Subred, and Zona DNS ... The 'Enrutamiento de red' (Network routing) section is partially visible at the bottom.

*Figura 62. Configuración de redes.*

## Protección de Datos

- **Habilitar la restauración a un momento dado para contenedores:** Permite restaurar contenedores a un estado anterior, protegiendo los datos de modificaciones o eliminaciones accidentales.

- **Habilitar la eliminación temporal para blobs y contenedores:** Permite recuperar los blobs o contenedores que se han marcado para su eliminación, lo que proporciona una capa adicional de protección contra eliminaciones accidentales.
- **Habilitar el control de versiones para blobs:** Habilita el control de versiones para los blobs, lo que permite mantener automáticamente las versiones anteriores de los blobs. Esto es útil para el seguimiento de cambios y la recuperación de versiones anteriores.
- **Habilitar la fuente de cambios del blob:** Realiza un seguimiento de la creación, modificación y eliminación de blobs dentro de la cuenta.
- **Habilitar la compatibilidad con la inmutabilidad de nivel de versión:** Permite establecer una directiva de retención con una duración definida a nivel de cuenta, la cual se aplica a todas las versiones de los blobs.

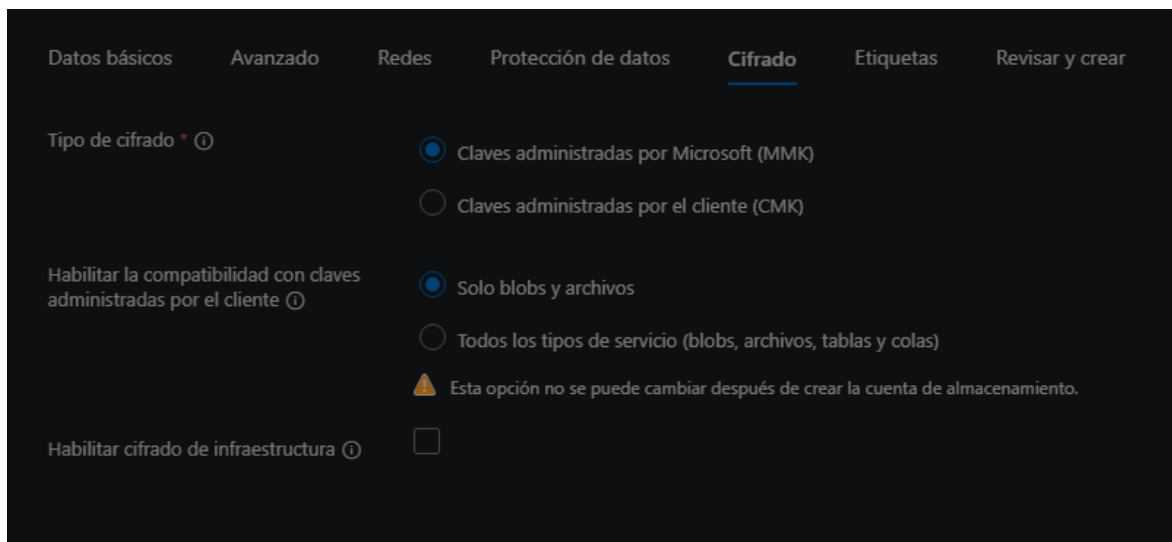
The screenshot shows the 'Protección de datos' (Data Protection) tab in the Azure portal. Under the 'Recuperación' (Recovery) section, there are three settings:

- Habilitar la restauración a un momento dado para contenedores** (Unchecked): Use the restore to a given moment to restore one or more containers to a previous state. If the restore to a given moment is enabled, version control, the source of changes, and the temporal deletion of blobs must also be enabled. [Obtener más información](#)
- Habilitar la eliminación temporal para blobs** (Checked): La eliminación temporal permite recuperar los blobs que se marcaron previamente para su eliminación, incluidos los blobs que se sobrescribieron. [Obtener más información](#)  
Días durante los cuales se conservarán los blobs eliminados: 7
- Habilitar la eliminación temporal para contenedores** (Checked): La eliminación temporal permite recuperar contenedores que se marcaron anteriormente para su eliminación. [Obtener más información](#)  
Habilitar la eliminación temporal para los datos que se sobrescriben con frecuencia puede resultar en un aumento de los costes de almacenamiento. [Obtener más información](#)  
Días durante los cuales se conservarán los contenedores eliminados: 7
- Habilitar la eliminación temporal para recursos compartidos de archivos** (Checked): La eliminación temporal permite recuperar los recursos compartidos de archivos que se marcaron previamente para su eliminación. [Obtener más información](#)  
Días durante los cuales se conservarán los recursos compartidos de archivos eliminados: 7

*Figura 63. Configuración de protección de datos.*

## Cifrado

- **Tipo de cifrado:** Azure ofrece la opción de habilitar el cifrado administrado por el cliente, lo que permite que las claves de cifrado sean gestionadas por el cliente, o usar el cifrado de infraestructura, que es gestionado automáticamente por Azure.
- **Habilitar cifrado de infraestructura:** Esta opción habilita el cifrado automático de todos los datos almacenados dentro de la cuenta utilizando claves gestionadas por **Azure**.



The screenshot shows the 'Cifrado' (Encryption) tab in the Azure portal. It features three main settings:

- Tipo de cifrado \*** (Encryption type): Two radio button options. 'Claves administradas por Microsoft (MMK)' (Managed by Microsoft keys) is selected, and 'Claves administradas por el cliente (CMK)' (Managed by customer keys) is unselected.
- Habilitar la compatibilidad con claves administradas por el cliente** (Enable compatibility with customer-managed keys): Two radio button options. 'Solo blobs y archivos' (Only blobs and files) is selected, and 'Todos los tipos de servicio (blobs, archivos, tablas y colas)' (All service types) is unselected. A warning icon and text below state: 'Esta opción no se puede cambiar después de crear la cuenta de almacenamiento.' (This option cannot be changed after creating the storage account).
- Habilitar cifrado de infraestructura** (Enable infrastructure encryption): An unchecked checkbox.

*Figura 64. Configuración de cifrado.*

## Etiquetas

Al igual que con otros recursos en Azure, podemos agregar etiquetas a la cuenta de almacenamiento para facilitar la organización y gestión de los recursos dentro de nuestra suscripción.

5. **Revisar y crear:** Después de ingresar toda la información necesaria, revisamos la configuración de la cuenta de almacenamiento y hacemos clic en "**Revisar + Crear**". Finalmente, seleccionamos "**Crear**" para desplegar la cuenta de almacenamiento.

Una vez creada la cuenta de almacenamiento, podemos utilizarla para almacenar diferentes tipos de datos, como archivos, contenedores de blobs, discos virtuales y más. El Almacenamiento de Blobs es uno de los usos más comunes, ya que nos permite almacenar grandes cantidades de datos no estructurados como imágenes, videos, backups y archivos.

Además, Azure Storage ofrece otras funcionalidades como la replicación geográfica para garantizar la disponibilidad y durabilidad de los datos, así como opciones de gestión de acceso para controlar quién tiene permiso para acceder a los recursos almacenados.



## Configuración de Seguridad de la Nube en Azure

La seguridad en la nube es un aspecto crítico para proteger tanto los datos como los servicios desplegados en Azure. Dependiendo del servicio, Azure ofrece configuraciones de seguridad predeterminadas y también permite personalizar diversas opciones de seguridad. A continuación, se explica cómo se maneja la seguridad en los servicios más comunes como Azure Container Registry, App Service y Azure Database for MySQL.

### 1. Seguridad en Azure Container Registry (ACR)

En Azure Container Registry, las opciones de seguridad están diseñadas para proteger tanto las imágenes de los contenedores como las credenciales de acceso al registro. Algunas configuraciones de seguridad clave son:

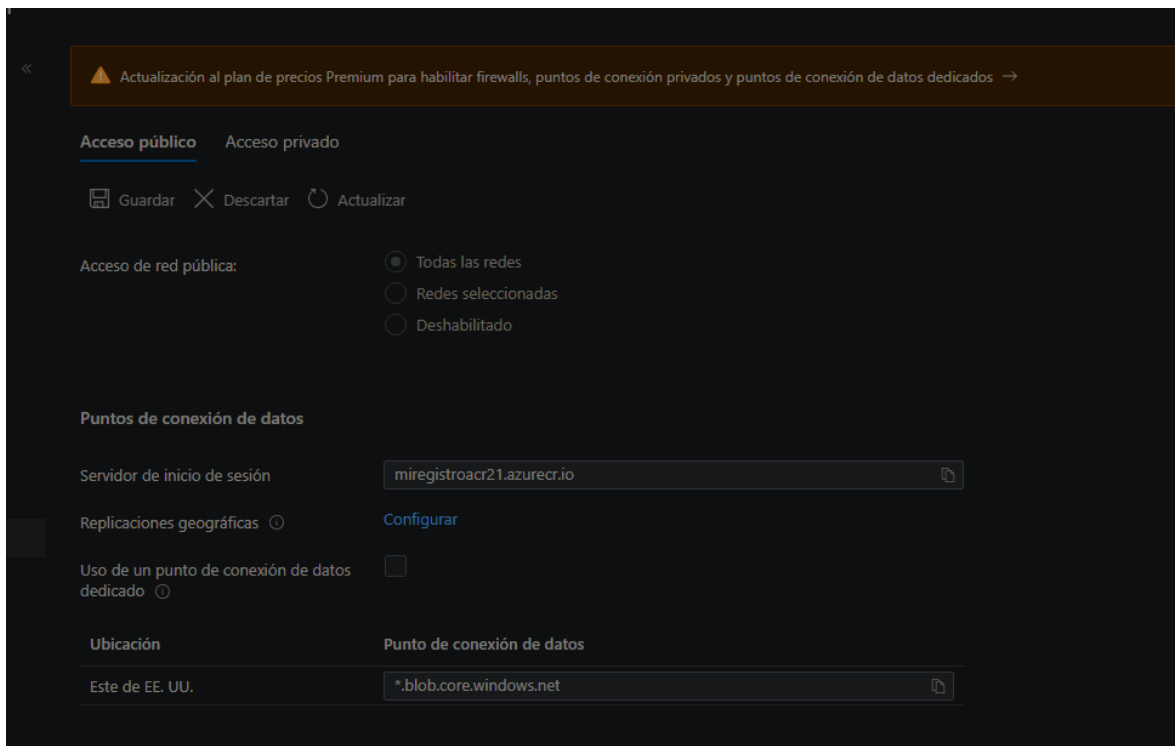
- **Cifrado en reposo:** Azure Container Registry cifra automáticamente las imágenes y otros artefactos cuando se almacenan en el registro. Este cifrado se realiza de manera predeterminada, utilizando el algoritmo AES-256. Además, es posible habilitar el cifrado mediante claves administradas por el cliente para un control más detallado de la seguridad.

El cifrado del servicio Azure Container Registry protege los datos en reposo. Azure Container Registry cifra las imágenes y otros artefactos cuando se insertan en el registro y los descifra automáticamente al extraerlos.

Los datos del registro de contenedor se cifran de forma predeterminada. Puede elegir usar el servicio Bring Your Own Key para cifrar las instancias de Azure Container Registry.

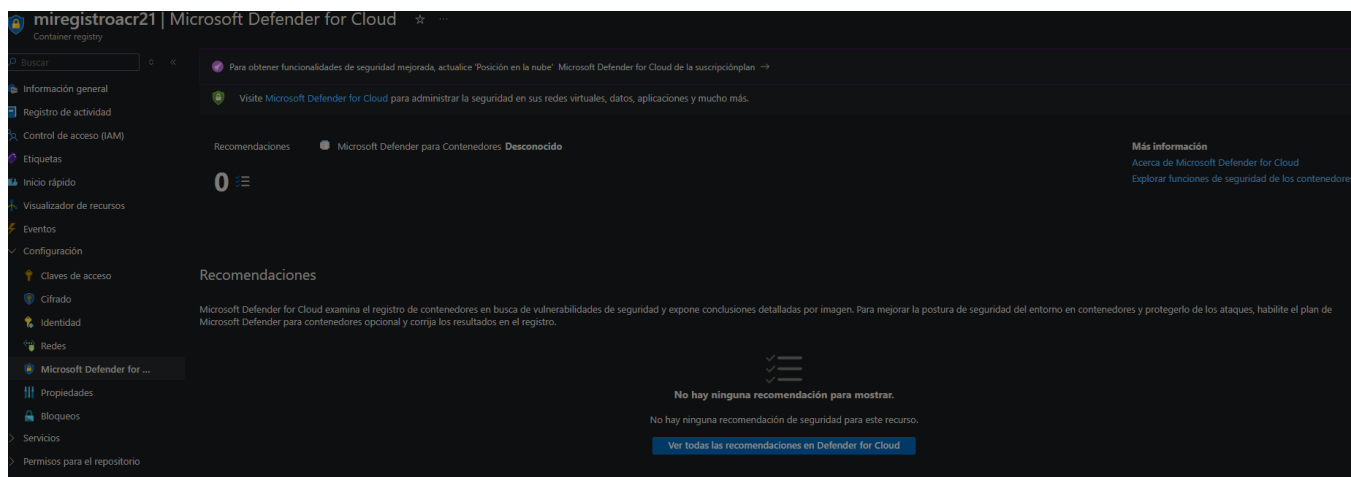
*Figura 65. Cifrado usado generalmente en los servicios de Azure.*

- **Control de acceso:** Se pueden definir políticas de acceso mediante Azure Active Directory (Azure AD) y Control de acceso basado en roles (RBAC) para gestionar quién tiene acceso al registro. Las credenciales de administrador y el uso de identidad gestionada también están disponibles para una gestión más segura de las imágenes.
- **Redes:** El acceso a Azure Container Registry puede restringirse mediante puntos de conexión privados y Redes Virtuales (VNET) para asegurar que el acceso se haga solo desde redes confiables. Sin embargo para implementar estas reglas se debe tener un plan premium.



*Figura 66. Configuración de redes en el container registry.*

- **Microsoft Defender for Cloud:** Podemos habilitar la protección de Microsoft Defender for Cloud para añadir una capa adicional de seguridad mediante el análisis de vulnerabilidades y la detección de amenazas en las imágenes almacenadas.



*Figura 67. Protección de Microsoft Defender Cloud.*

## 2. Seguridad en App Service

**App Service** es una plataforma altamente flexible que permite desplegar aplicaciones web y APIs. Azure proporciona varias características de seguridad integradas y opciones para personalizar según nuestras necesidades:

- **Cifrado de datos en reposo y en tránsito:** **App Service** asegura que todos los datos estén cifrados tanto en reposo como en tránsito. Utiliza **TLS (Transport Layer Security)** para las comunicaciones seguras entre la aplicación y los usuarios. Además, la encriptación en reposo está habilitada por defecto para las bases de datos y otros recursos asociados a la aplicación.
- **Redes:** El **App Service** puede configurarse para que la aplicación sea pública o privada. Si seleccionamos una red privada, se puede acceder a la aplicación a través de un punto de conexión privado y configurando las redes virtuales (VNETs). Esto proporciona un control total sobre quién puede acceder a la aplicación.

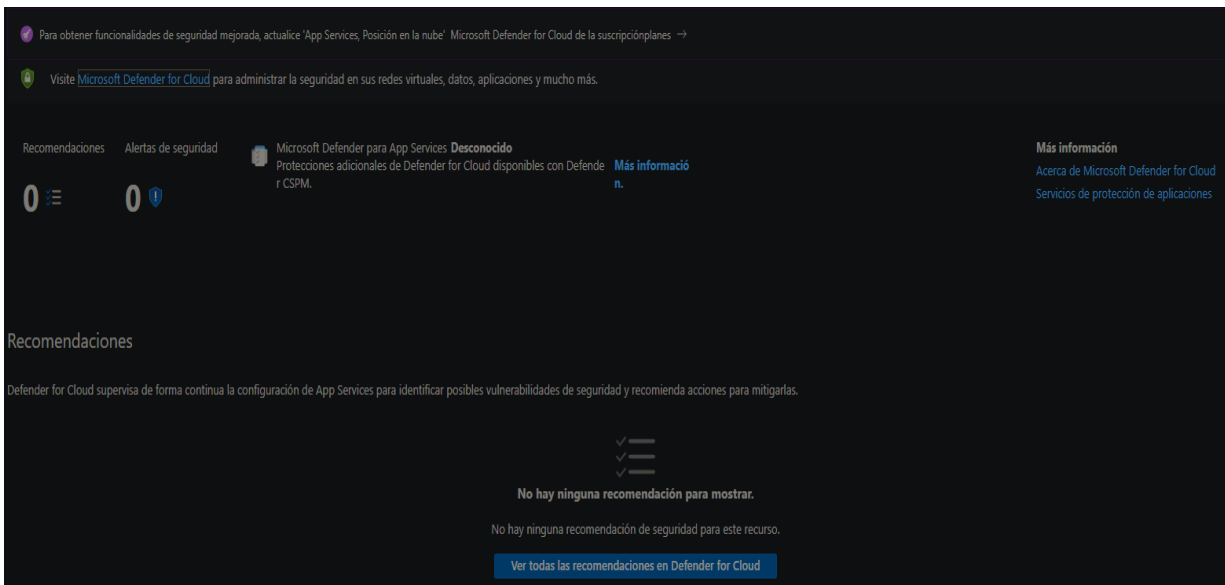
Compruebe la configuración de red. Seleccione cualquiera de las características que se enumeran a continuación para cambiar la configuración de red. [Más información](#)

Configuración del tráfico entrante		Configuración del tráfico saliente	
Acceso a la red pública	Habilitado sin restricciones de acceso	Integración de red virtual	Sin configurar
Dirección asignada a la aplicación	Sin configurar	Conexiones híbridas	Sin configurar
Puntos de conexión privados	0 puntos de conexión privados	DNS de salida	Predeterminado (proporcionado por Azure)
Direcciones de entrada	20.48.204.4	Direcciones de salida	20.175.198.203, 20.175.198.211, 20.175... <a href="#">Mostrar más</a>
Servicios de entrada opcionales		Configuración de subred de integración	
Azure Front Door	<a href="#">Ver detalles</a>	NAT Gateway	N/A
		Grupo de seguridad de red	N/A
		Ruta definida por el usuario	N/A

*Figura 68. Configuración de redes en App Services.*

- **Autenticación y autorización:** **Azure App Service** permite integrar la autenticación con **Azure Active Directory (Azure AD)** o servicios de terceros, lo que garantiza que solo los usuarios autorizados puedan acceder a las aplicaciones. Además, se pueden configurar **reglas de seguridad** mediante **grupos de seguridad de red (NSG)** para proteger los recursos.

- **Microsoft Defender for Cloud:** Se puede activar la integración con **Microsoft Defender for Cloud** para supervisar y proteger los servicios, detectando posibles amenazas y vulnerabilidades.



*Figura 69. Protección de Microsoft Defender Cloud para app services.*

De igual manera, App Service ofrece certificados administrados gratuitos que son completamente gestionados por Azure para garantizar la seguridad del sitio. Estos certificados se usan principalmente para HTTPS y están destinados a mantener la seguridad a un nivel máximo.

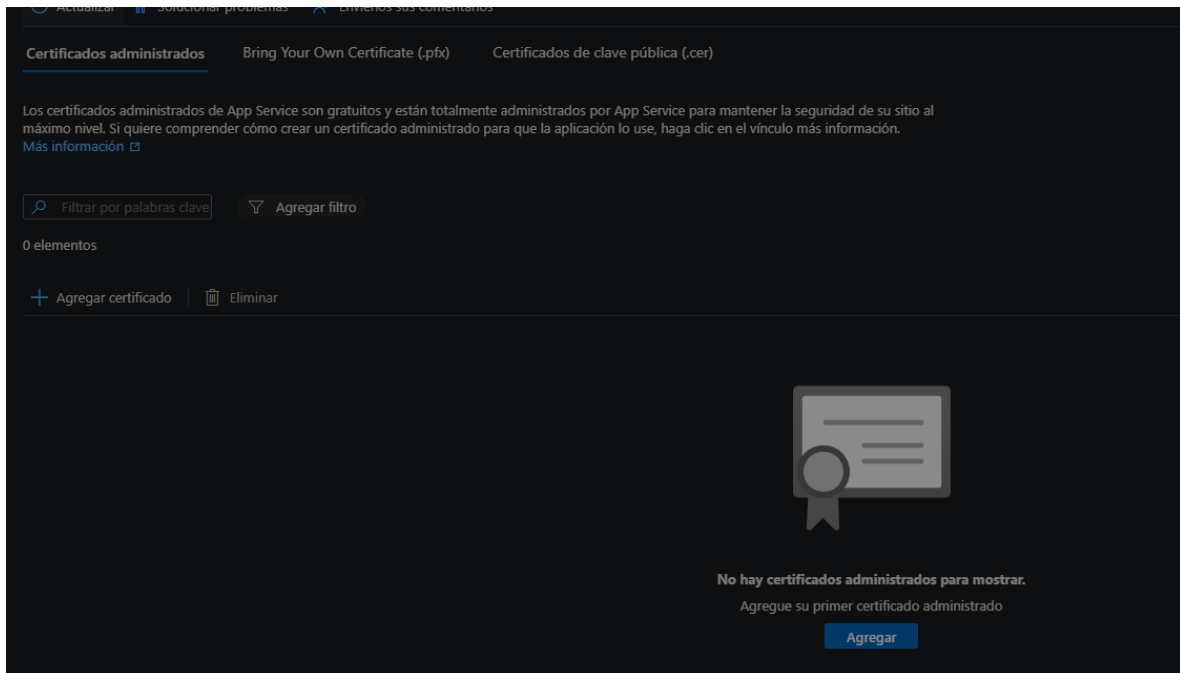
- Estos certificados son fáciles de implementar y configurar, y no requieren intervención manual para renovaciones o mantenimiento, ya que **Azure** gestiona todo el ciclo de vida del certificado.

#### **Certificados Personalizados (Bring Your Own Certificate - BYOC):**

- Si se prefiere usar un certificado personalizado, como uno comprado a una autoridad certificadora externa, **Azure** permite que los usuarios suban su propio certificado en formato **.pfx** o **.cer**.
- Los certificados **.pfx** se utilizan cuando se requiere una clave privada junto con el certificado, mientras que los **.cer** son usados cuando solo se requiere el certificado público.
- Estos certificados pueden ser cargados en el portal de **App Service** y asignados a la aplicación web para su uso.

#### **Proceso para Agregar Certificados Administrados:**

- Dentro de la configuración de **App Service**, seleccionamos la opción "**Certificados administrados**" y luego podemos hacer clic en "**Agregar certificado**" para subir un nuevo certificado o utilizar uno proporcionado por **Azure**.
- Si no hay certificados previamente cargados, se muestra un mensaje indicativo, tal como "**No hay certificados administrados para mostrar**", y se nos da la opción de agregar uno por primera vez.

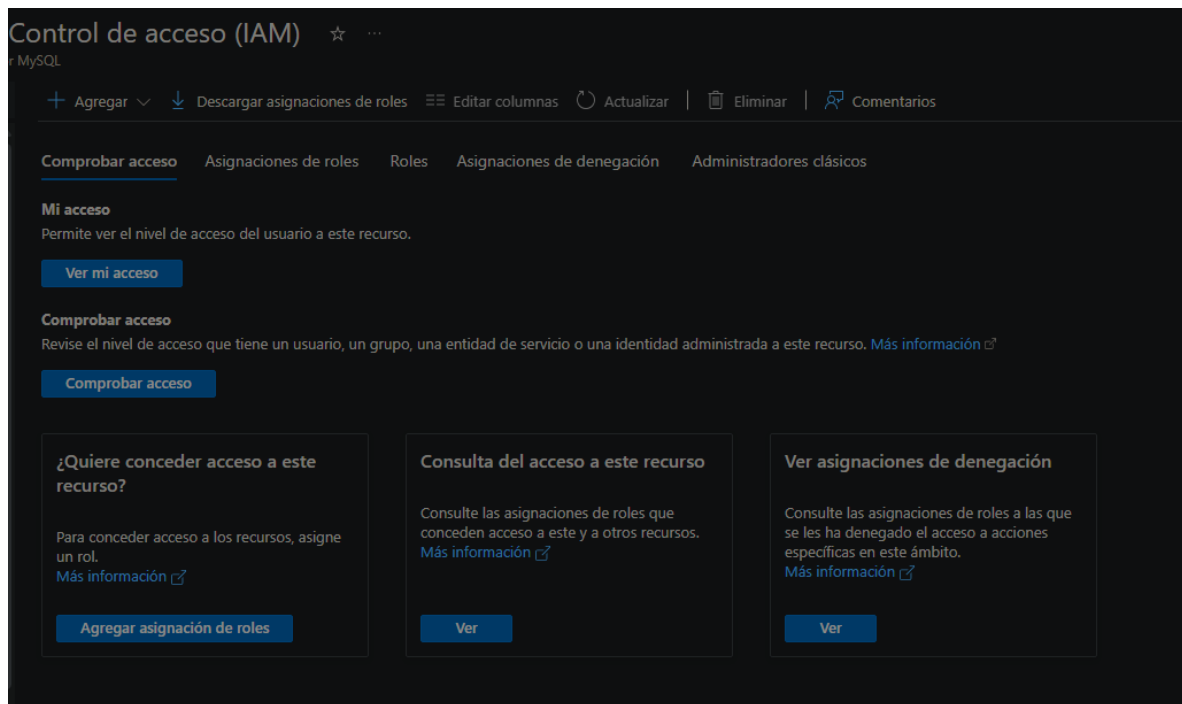


*Figura 70. Certificados utilizados en el App Services.*

### 3. Seguridad en Azure Database for MySQL

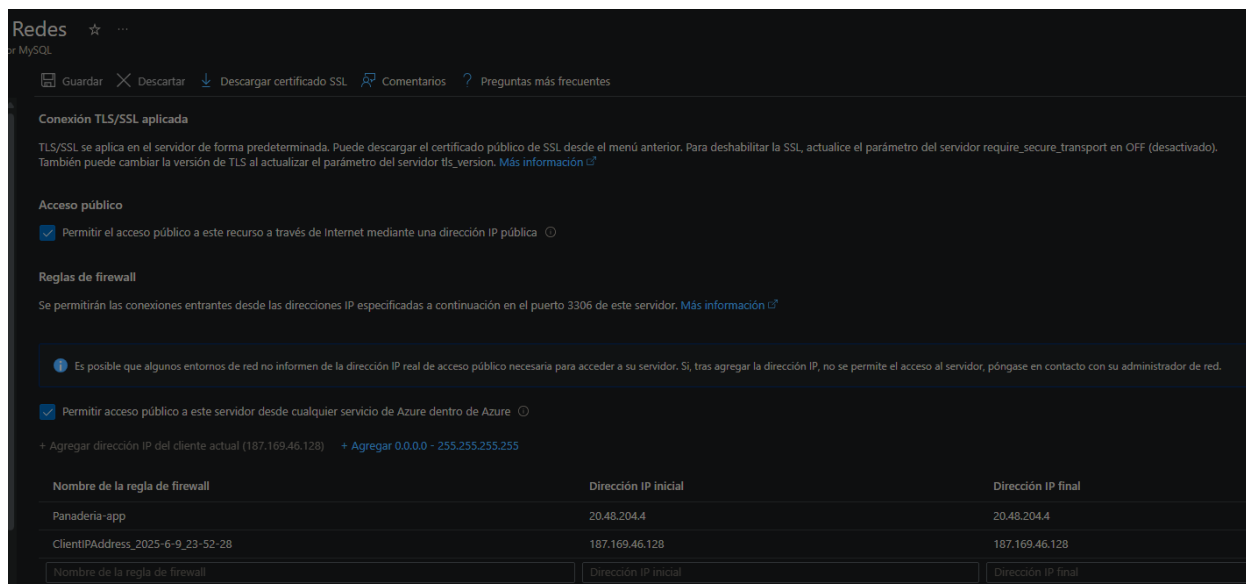
La seguridad en Azure Database for MySQL es crucial, especialmente cuando se manejan datos sensibles. Azure proporciona varias configuraciones de seguridad tanto por defecto como personalizables para asegurar las bases de datos:

- **Cifrado de datos en reposo:** Azure Database for MySQL cifra automáticamente los datos almacenados utilizando **AES-256**. Esto es transparente y no requiere configuración adicional. Los datos en reposo se encuentran protegidos por defecto, aunque se puede configurar un cifrado gestionado por el cliente para un mayor control sobre las claves.
- **Autenticación segura:** Azure Database for MySQL permite utilizar autenticación basada en Azure Active Directory o autenticación tradicional mediante usuario y contraseña. Además, se pueden configurar reglas de firewall para restringir las direcciones IP que pueden acceder a la base de datos, limitando el acceso solo a direcciones autorizadas.



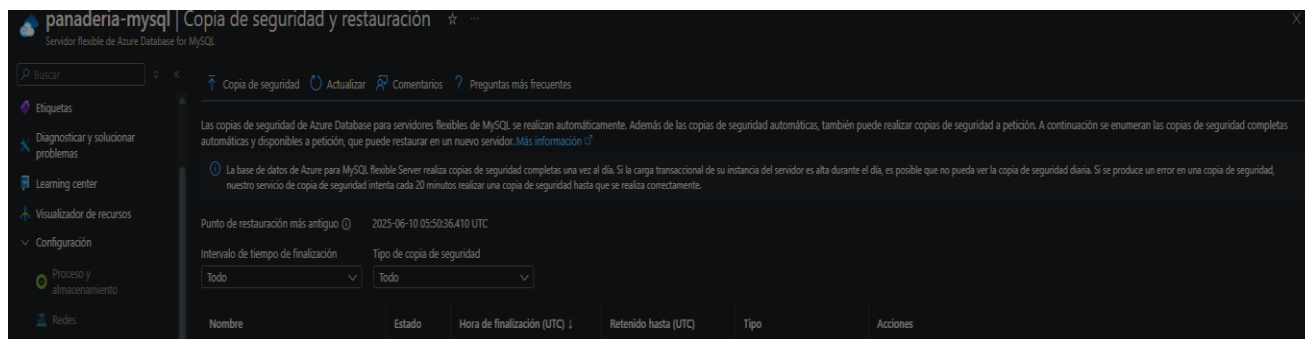
*Figura 71. Control de acceso (IAM).*

- **Redes:** En Azure Database for MySQL, se puede habilitar el acceso público a través de una dirección IP pública o configurar el acceso privado mediante puntos de conexión privados. El acceso público se gestiona mediante reglas de firewall, donde se pueden permitir conexiones desde direcciones IP específicas, como por ejemplo, desde direcciones internas de Azure. Si se requiere mayor seguridad, se puede restringir el acceso público y habilitar el acceso privado a la base de datos a través de una red virtual (VNET), lo que garantiza que la base de datos esté aislada y disponible solo para los recursos dentro de la misma red. Además, la conexión TLS/SSL está habilitada de manera predeterminada, asegurando que los datos en tránsito estén protegidos, y es posible configurar la versión de TLS o deshabilitar la conexión SSL si es necesario.



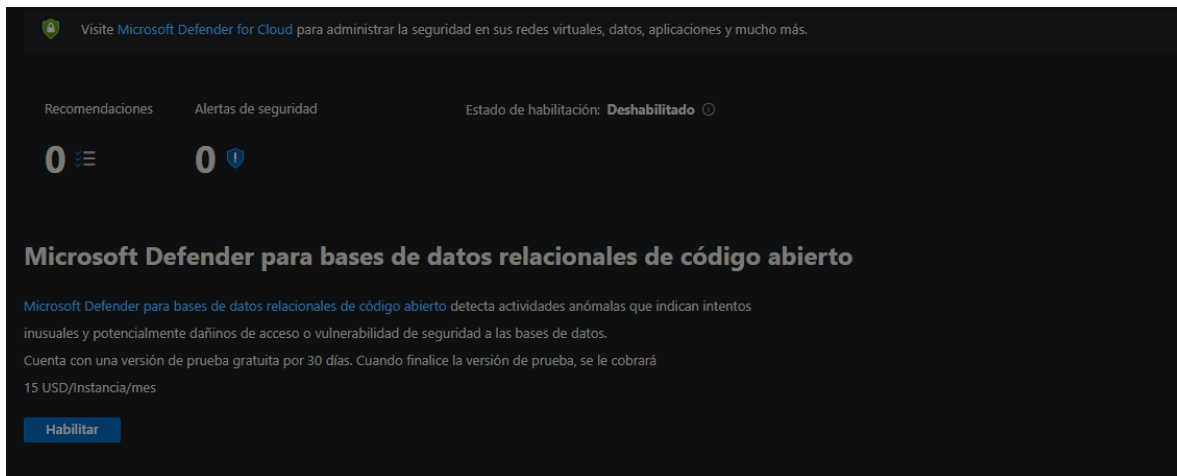
*Figura 72. Configuración de redes para la BD en Azure.*

- **Respaldo y recuperación:** Azure ofrece opciones de copia de seguridad automáticas y la posibilidad de restaurar bases de datos a un momento específico, asegurando que los datos no se pierdan ante una eliminación accidental o un ataque.



*Figura 73. Apartado de copias de seguridad y restauración de la BD.*

- **Microsoft Defender for Cloud:** También se puede habilitar **Defender for Cloud** para recibir recomendaciones de seguridad y monitoreo de las amenazas, así como para proteger contra accesos no autorizados.



*Figura 74. Protección de Microsoft Defender Cloud para la BD.*

Cabe mencionar que aunque **Microsoft Defender for Cloud** ofrece una prueba gratuita de 30 días, el acceso a todas sus funciones avanzadas tiene un costo adicional. Durante la prueba gratuita, los usuarios pueden explorar las funcionalidades de seguridad avanzada, como la detección de amenazas, evaluaciones de seguridad y recomendaciones de optimización de la infraestructura, pero una vez finalizado el periodo de prueba, se aplicarán tarifas dependiendo del tipo de suscripción y los servicios utilizados.



## Conclusión

En conclusión, la práctica realizada sobre la configuración y despliegue de microservicios en la nube, utilizando plataformas como AWS, Azure y GCP, ha sido una experiencia enriquecedora que nos permitió fortalecer nuestras habilidades técnicas y adquirir una comprensión más profunda de los sistemas distribuidos. A través de esta práctica en parejas, no solo pudimos desarrollar y desplegar un microservicio utilizando Flask en una instancia virtual de AWS, sino que también aprendimos a gestionar y configurar los entornos necesarios para su funcionamiento adecuado.

El trabajo en la nube, utilizando modelos como IaaS, PaaS y SaaS, nos brindó la oportunidad de comparar las ventajas y características de cada proveedor, destacando tanto las similitudes como las diferencias clave en cuanto a su facilidad de uso, herramientas de administración y enfoque en la seguridad. Nos enfrentamos a la configuración de recursos virtuales, la gestión de redes, y la integración de medidas de seguridad, lo cual fue crucial para comprender cómo las plataformas en la nube facilitan la creación de soluciones escalables y seguras.

Además, el monitoreo de los recursos y el uso de herramientas como CloudWatch, Azure Monitor y Cloud Logging nos ayudaron a entender la importancia de una administración eficiente, lo que es fundamental en el contexto de sistemas distribuidos. De esta manera, la práctica no solo fortaleció nuestras competencias técnicas, sino que también nos preparó para tomar decisiones informadas y estratégicas al elegir un proveedor en función de las necesidades de un proyecto.

Por último, cabe mencionar que, esta experiencia ha sido esencial para comprender la relevancia de la infraestructura en la nube en el desarrollo de aplicaciones modernas y cómo las plataformas de la nube son fundamentales en la creación de soluciones distribuidas y escalables. Sin duda, nos ha proporcionado una ventaja competitiva al adquirir conocimientos clave sobre las tecnologías que dominan la industria de los sistemas distribuidos.

## Referencias

- Amazon Web Services. (s.f.). Cloud Computing – Servicios de informática en la nube. <https://aws.amazon.com/es/>
- Mendieta, A. (17 de marzo de 2024). Qué es AWS: Un mundo de soluciones en la nube. Open Webinars. <https://openwebinars.net/blog/que-es-aws/>
- Microsoft Azure. (s.f.). Servicios de informática en la nube. <https://azure.microsoft.com/es-es>
- Google Cloud Platform. (s.f.). <https://cloud.google.com>
- Google Developers. (2024). Getting started with GCP. <https://developers.google.com>