**INSTITUTO POLITÉCNICO NACIONAL**

ESCUELA SUPERIOR DE CÓMPUTO

**UNIDAD DE APRENDIZAJE**

SISTEMAS DISTRIBUIDOS

**PRACTICA No. 9**

CONFIGURACION DE NUBES

**ALUMNO**

GÓMEZ GALVAN DIEGO YAEL

**GRUPO**

7CM1

**PROFESOR**

CARRETO ARELLANO CHADWICK

**FECHA DE ENTREGA**

05 DE JUNIO DE 2025

**INDICE**

[ANTESCEDENTES 3](#_Toc200578241)

[Modelos de Servicio en la Nube 3](#_Toc200578242)

[Modelos de Implementación de la Nube 4](#_Toc200578243)

[Principales Proveedores de Nube Pública 4](#_Toc200578244)

[Amazon Web Services (AWS) 5](#_Toc200578245)

[Microsoft Azure 5](#_Toc200578246)

[Google Cloud Platform (GCP) 6](#_Toc200578247)

[PLANTEAMIENTO DEL PROBLEMA 7](#_Toc200578248)

[PROPUESTA DE SOLUCIÓN 8](#_Toc200578249)

[MATERIALES Y METODOS 9](#_Toc200578250)

[DESARROLLO DE SOLUCIÓN 10](#_Toc200578251)

[Google Cloud Platform (GCP) 10](#_Toc200578252)

[Creación y estructuración del proyecto 10](#_Toc200578253)

[Activación de APIs necesarias 11](#_Toc200578254)

[Configuración de seguridad 14](#_Toc200578255)

[Configuración de almacenamiento 16](#_Toc200578256)

[Configuración de monitoreo y logging 17](#_Toc200578257)

[Microsoft Azure 18](#_Toc200578258)

[Creación de cuenta 18](#_Toc200578259)

[Crear grupo de recursos 19](#_Toc200578260)

[Crear cuenta de almacenamiento 19](#_Toc200578261)

[Configurar contenedores dentro del almacenamiento 20](#_Toc200578262)

[Configurar control de acceso 20](#_Toc200578263)

[Crear registro de contenedores 21](#_Toc200578264)

[Crear recurso Application Insights 22](#_Toc200578265)

[RESULTADOS 23](#_Toc200578266)

[CONCLUSIONES 25](#_Toc200578267)

# **ANTESCEDENTES**

La computación en la nube es un modelo tecnológico que permite acceder, gestionar y consumir recursos informáticos a través de Internet, sin necesidad de contar con infraestructura física local. En lugar de depender de servidores propios o centros de datos internos, las organizaciones utilizan servicios proporcionados por proveedores externos, quienes alojan y administran estos recursos en centros de datos distribuidos a nivel global.

El concepto de computación en la nube comenzó a surgir a finales de los años 90, influenciado por avances en redes, virtualización y la expansión del modelo cliente-servidor. Su adopción masiva inició en la década del 2000 con el lanzamiento de Amazon Web Services (AWS) en 2006, marcando un hito que cambió para siempre la forma en que se concibe la infraestructura tecnológica. Desde entonces, la nube ha evolucionado hasta convertirse en un componente esencial de la transformación digital en todo tipo de organizaciones, desde startups hasta grandes corporativos.

La nube ofrece acceso bajo demanda a recursos como potencia de cómputo (servidores virtuales), almacenamiento, bases de datos, redes, plataformas de desarrollo, análisis de datos e incluso inteligencia artificial, sin necesidad de grandes inversiones iniciales ni de mantenimiento especializado.

Una de las principales razones del crecimiento exponencial de la computación en la nube es su escalabilidad dinámica, es decir, la capacidad de ajustar recursos automáticamente según la demanda. Esta característica, junto con su alta disponibilidad, resiliencia ante fallos, y su modelo económico basado en el consumo real ("*pago por uso*"), ha permitido a individuos, empresas e instituciones optimizar recursos, reducir costos y acelerar el desarrollo de soluciones tecnológicas.

El uso de tecnologías clave como la virtualización, los contenedores, la automatización mediante APIs, la infraestructura como código (IaC) y los entornos DevOps han potenciado aún más las capacidades de la nube, permitiendo a los usuarios crear y administrar entornos complejos con rapidez, control y eficiencia.

Según Gartner (2024), más del 95% de las nuevas cargas de trabajo digitales serán desplegadas en plataformas en la nube para 2025, lo que refleja su papel protagónico en la estrategia tecnológica global.

## **Modelos de Servicio en la Nube**

La computación en la nube se organiza principalmente en tres modelos de servicio, cada uno con distintos niveles de control, responsabilidad y abstracción:

* **IaaS (Infraestructura como Servicio):** Es el modelo más básico de los tres. Este proporciona acceso a recursos fundamentales como máquinas virtuales, redes, almacenamiento y sistemas operativos. Aquí el usuario es responsable de administrar el sistema operativo, las aplicaciones y los datos, mientras que el proveedor se encarga del hardware subyacente. Este modelo es ideal para administradores de sistemas que requieren flexibilidad y control sobre la infraestructura.
* **PaaS (Plataforma como Servicio):** Este modelo ofrece un entorno listo para desarrollar, probar y desplegar aplicaciones, sin preocuparse por la gestión de servidores, redes o almacenamiento. Es muy útil para desarrolladores, ya que permite concentrarse en el código y la lógica de negocio.
* **SaaS (Software como Servicio):** En este modelo, los usuarios acceden a aplicaciones totalmente funcionales desde un navegador o cliente, sin instalar nada localmente ni gestionar la infraestructura. Este modelo es el más utilizado por el usuario final y empresas que buscan herramientas listas para operar.

## **Modelos de Implementación de la Nube**

Además de los modelos de servicio, la nube también puede clasificarse según su forma de implementación:

* **Nube pública:** Los servicios son proporcionados por un proveedor externo a través de Internet y son compartidos por múltiples organizaciones. Este modelo es el más accesible y escalable, ideal para proyectos dinámicos o que requieren gran elasticidad.
* **Nube privada:** La infraestructura es usada exclusivamente por una sola organización. Esta puede estar ubicada en las instalaciones del cliente o ser gestionada por un proveedor externo, de esta manera permite ofrecer un mayor control, seguridad y personalización, siendo adecuada para empresas con requerimientos regulatorios o de privacidad específicos.
* **Nube híbrida:** Combina nubes públicas y privadas, permitiendo la movilidad de datos y aplicaciones entre ambas. Este enfoque ofrece una mayor flexibilidad y optimización de recursos, permitiendo mantener cargas sensibles en la nube privada y escalar otras en la nube pública.

## **Principales Proveedores de Nube Pública**

Con el crecimiento acelerado del cómputo en la nube, diversos proveedores han desarrollado plataformas especializadas que ofrecen servicios escalables y confiables a organizaciones de todo el mundo. Entre ellos, destacan tres gigantes tecnológicos que lideran el mercado global de servicios en la nube: Amazon Web Services (AWS), Microsoft Azure y Google Cloud Platform (GCP). Cada uno presenta un conjunto robusto de herramientas, servicios y recursos que se adaptan a las necesidades de diferentes tipos de usuarios y sectores, los cuales se abordaran a continuación:

### **Amazon Web Services (AWS)**

Amazon Web Services, lanzado oficialmente en 2006 por Amazon, es ampliamente reconocido como el pionero en el sector de la computación en la nube. Gracias a su madurez, robustez y constante innovación, AWS ha mantenido una posición de liderazgo en el mercado global.

Su catálogo incluye más de 200 servicios distribuidos en categorías como cómputo, almacenamiento, bases de datos, redes, herramientas de desarrollo, análisis de datos, aprendizaje automático, seguridad, inteligencia artificial e Internet de las cosas (IoT).

Entre sus servicios más representativos destacan:

* **Amazon EC2 (Elastic Compute Cloud):** Provisión de máquinas virtuales configurables bajo demanda.
* **Amazon S3 (Simple Storage Service):** Almacenamiento escalable de objetos.
* **Amazon RDS (Relational Database Service):** Gestión de bases de datos relacionales como MySQL, PostgreSQL o SQL Server.
* **AWS Lambda:** Ejecución de funciones sin necesidad de administrar servidores (arquitectura serverless).

AWS opera a través de una red global de regiones y zonas de disponibilidad, lo que garantiza alta disponibilidad, redundancia y baja latencia. Además, brinda herramientas sólidas para automatización (como AWS CloudFormation), integración continua, monitoreo, gestión de identidades y cumplimiento normativo.

### **Microsoft Azure**

Microsoft Azure fue lanzado en 2010 como parte de la estrategia de Microsoft para consolidarse en el mundo de los servicios en la nube. Su integración natural con el ecosistema de Microsoft ha sido clave para su adopción por parte de empresas que ya utilizan soluciones como Windows Server, Active Directory, Microsoft 365, SQL Server y Visual Studio.

Azure ofrece una amplia gama de servicios de IaaS, PaaS y SaaS, y es particularmente fuerte en entornos híbridos, facilitando la migración gradual de aplicaciones locales hacia la nube. Entre sus servicios más relevantes se encuentran:

* **Azure Virtual Machines:** Máquinas virtuales configurables con distintos sistemas operativos.
* **Azure App Services:** Plataforma para desplegar aplicaciones web y APIs.
* **Azure SQL Database:** Base de datos como servicio basado en SQL Server.
* **Azure DevOps y GitHub Actions:** Herramientas para CI/CD y gestión de proyectos de desarrollo.
* **Microsoft Defender y Azure Sentinel:** Soluciones avanzadas de seguridad y detección de amenazas.

Azure está diseñado bajo principios de confianza cero, con controles de seguridad avanzados, cumplimiento con múltiples normativas internacionales y opciones de gestión granular de identidades y accesos (Azure Active Directory). También ofrece compatibilidad con soluciones de código abierto y tecnologías como Kubernetes y contenedores Docker.

### **Google Cloud Platform (GCP)**

Google Cloud Platform, lanzada en 2011, es la propuesta de Google para ofrecer servicios en la nube orientados al alto rendimiento, la escalabilidad y el análisis de datos. Si bien su adopción fue más gradual que la de sus competidores, GCP ha ganado terreno debido a su infraestructura altamente eficiente y su liderazgo en tecnologías como Big Data, aprendizaje automático y contenedores.

Entre sus servicios más destacados se encuentran:

* **Compute Engine:** Máquinas virtuales de alto rendimiento.
* **Cloud Storage:** Almacenamiento seguro y escalable de objetos.
* **BigQuery:** Plataforma de análisis de grandes volúmenes de datos en tiempo real.
* **Cloud Functions:** Ejecución de funciones en entornos serverless.
* **Kubernetes Engine (GKE):** Servicio gestionado de orquestación de contenedores basado en Kubernetes.

Una de las principales ventajas de GCP es su uso de la misma infraestructura global que Google emplea para productos como Gmail, YouTube y Google Search, lo que garantiza velocidad, disponibilidad y confiabilidad a gran escala. Además, su enfoque en inteligencia artificial y ciencia de datos lo convierte en una plataforma muy atractiva para startups y organizaciones centradas en la innovación.

# **PLANTEAMIENTO DEL PROBLEMA**

En el entorno tecnológico actual, el uso de infraestructura física para alojar servicios informáticos presenta importantes limitaciones para organizaciones de todos los tamaños. La adquisición y mantenimiento de servidores locales, el aprovisionamiento de espacio físico, la configuración manual de seguridad, así como la falta de escalabilidad inmediata, generan altos costos operativos, dependencia técnica y dificultades para responder de manera ágil a las nuevas demandas digitales.

A medida que los sistemas requieren mayor disponibilidad, flexibilidad y eficiencia, se vuelve insostenible depender exclusivamente de servidores tradicionales. Esto se acentúa aún más cuando se desea centralizar recursos, mejorar la seguridad, monitorear servicios en tiempo real o gestionar repositorios de forma remota.

Frente a este escenario, la configuración inicial de plataformas de computación en la nube representa un paso clave para garantizar una infraestructura moderna, escalable y eficiente. Proveedores como Amazon Web Services (AWS), Microsoft Azure y Google Cloud Platform (GCP) permiten a los usuarios realizar un aprovisionamiento inteligente de sus entornos tecnológicos desde el navegador, sin necesidad de equipos físicos ni instalaciones complejas.

Sin embargo, para aprovechar correctamente estas plataformas, es necesario realizar una configuración adecuada desde el inicio, que contemple:

* La creación y configuración de la cuenta o suscripción correspondiente.
* La optimización de la plataforma, ajustando parámetros clave como regiones, costos, rendimiento y alertas.
* La implementación de controles de seguridad básicos, como usuarios, roles, políticas y permisos.
* La creación de espacios de almacenamiento seguros y gestionables.
* El acceso a herramientas de monitoreo y métricas para supervisar el uso y comportamiento de los recursos.
* La activación de servicios fundamentales como cómputo, bases de datos y redes.
* La configuración de repositorios para la gestión del código fuente o proyectos futuros.

# **PROPUESTA DE SOLUCIÓN**

Como solución al problema que representa la dependencia de infraestructura física para la gestión de servicios tecnológicos, se propone la realización de la configuración básica de entornos en la nube pública utilizando tres plataformas líderes: Amazon Web Services (AWS), Microsoft Azure y Google Cloud Platform (GCP).

Esta propuesta consiste en establecer los elementos esenciales que permitan utilizar la nube de forma segura, eficiente y escalable, abarcando el registro de cuentas, la optimización y configuración inicial, la implementación de medidas de seguridad básicas, la creación de espacios de almacenamiento, la activación de servicios fundamentales, el monitoreo de recursos y la gestión de repositorios.

El objetivo general es contar con una infraestructura flexible y lista para futuras implementaciones de sistemas, eliminando la necesidad de servidores físicos y garantizando una base tecnológica moderna y adaptable.

# **MATERIALES Y METODOS**

Para la configuración inicial de entornos en la nube publica, se utilizaron diversas herramientas, plataformas y recursos organizados en torno a tres proveedores principales: Amazon Web Services (AWS), Microsoft Azure y Google Cloud Platform (GCP). Esta práctica se enfocó en preparar un entorno de trabajo optimizado y seguro en cada nube, como base para futuros despliegues tecnológicos.

**Materiales**

Para la realización de esta práctica de configuración básica de nubes públicas, se utilizaron los siguientes recursos:

* **Sistema Operativo**: Windows 11  
  Plataforma principal sobre la cual se accedieron a los portales de administración de cada proveedor de nube
* **Navegador Web:** Google Chrome (version 125)

Herramienta utilizada para el acceso y navegación dentro de las consolas de administración de AWS, Azure y GCP

* **Proveedor de nube:** Amazon Web Service (AWS)

Plataforma donde se realizó la creación de cuenta, activación de facturación, configuración de almacenamiento (S3), monitoreo (CloudWatch), servicios iniciales y repositorio (CodeCommit).

* **Proveedor de nube:** Microsoft Azure

Plataforma en la que se configuró la suscripción, habilitación de recursos básicos como almacenamiento (Blob Storage), seguridad (Azure Active Directory), monitoreo (Azure Monitor) y repositorios (Azure Repos).

* **Proveedor de nube:** Google Cloud Platform (GCP)

Plataforma en la que se realizó la activación del proyecto, configuración de almacenamiento (Cloud Storage), seguridad (IAM), monitoreo (Cloud Monitoring) y repositorio de código (Cloud Source Repositories).

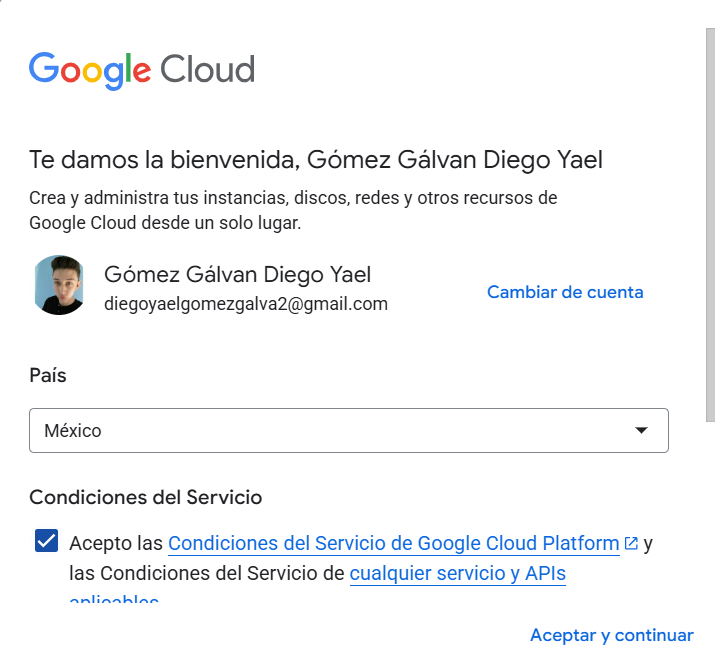
# **DESARROLLO DE SOLUCIÓN**

## **Google Cloud Platform (GCP)**

Para la implementación de las nubes, primeramente configuramos la nube Google Cloud Platform (GCP). Para ello realizamos la siguiente:

### **Creación y estructuración del proyecto**

Primeramente creamos una cuenta en la página oficial de GCP y llenamos la información solicitada respecto a Facturación.



Posteriormente, pasamos a la creación de un nuevo proyecto, que es donde próximamente realizaremos el despliegue de las practicas referente a una infraestructura basada en servicios web, en microservicios y una Progressive Web App (PWA), denominado como SistemasDistribuidos2025.

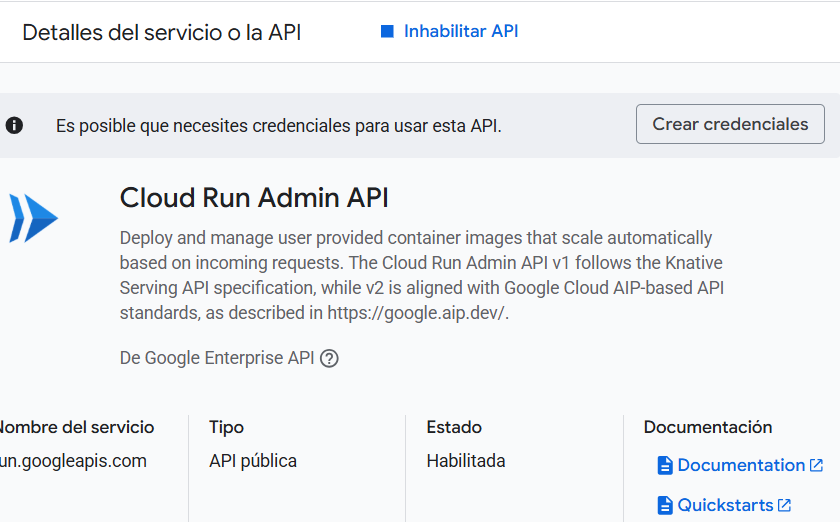
Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### **Activación de APIs necesarias**

Después de generar el proyecto, habilitamos las APIs que necesitaremos para realizar en el futuro realizar el despliegue de las practicas, las cuales son:

* Cloud Run API: Se habilita para permitir el despliegue y ejecución de aplicaciones en contenedores sin necesidad de administrar servidores. En este proyecto, se utilizará para alojar los microservicios del sistema bancario y el backend de la PWA, con escalado automático y acceso mediante HTTP/HTTPS.



* Cloud Build API: Se utiliza para automatizar la construcción y el despliegue de las aplicaciones. Será útil para compilar el código fuente, construir imágenes Docker de los microservicios y servicios web, y desplegarlos automáticamente en Cloud Run o App Engine, facilitando la integración de pipelines de CI/CD.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

* Artifact Registry API: Permite almacenar, gestionar y compartir imágenes de contenedores Docker y otros artefactos. En este caso, se usará como repositorio centralizado para guardar las imágenes Docker de los microservicios y servicios web antes de ser desplegados.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

* Cloud Storage API: Proporciona acceso al sistema de almacenamiento de archivos escalable de GCP. Se utilizará para almacenar archivos estáticos del frontend de la PWA (si no se usa Firebase Hosting), respaldos, bases de datos portables como SQLite o archivos compartidos entre servicios.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto, Correo electrónico

El contenido generado por IA puede ser incorrecto.

* App Engine API: Habilita el uso de App Engine para desplegar aplicaciones web sin necesidad de administrar infraestructura. En este proyecto, será utilizada para desplegar el backend del sistema de servicios web desarrollado con un framework, ofreciendo una plataforma simplificada para su ejecución.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

* Firebase Hosting API: Permite desplegar sitios web estáticos o progresivos (PWA) en una red global con certificado SSL. Se usará para publicar el frontend de la PWA (HTML, CSS, JS) de forma segura, rápida y moderna, optimizada para dispositivos móviles y navegadores.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

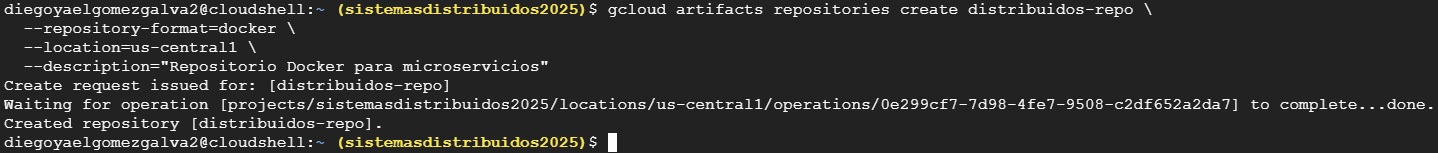
El contenido generado por IA puede ser incorrecto.

Ya con las APIs habilitadas, generaremos un repositorio de contenedores, que será donde se almacenaran las imágenes Dockers de los servicios. Para ello desde la terminal ejecutaremos el comando *gcloud ini*, el cual nos solicitara la cuenta que estamos usando, el proyecto y la región, tal como se muestra en la siguiente imagen.

Texto

El contenido generado por IA puede ser incorrecto.

Posteriormente, creamos el repositorio llamado “Distribuidos-Repo” cuya descripción es “Repositorio Docker para microservicios”



Por último, para concluir esta parte de la configuración, ejecute el comando *gcloud auth configure-docker us-central1-docker.pkg.dev* para que pueda autenticarse automáticamente con Artifact Registry usando las credenciales de gcloud.

Texto

El contenido generado por IA puede ser incorrecto.

### **Configuración de seguridad**

Durante esta fase se realizó la configuración básica de seguridad en Google Cloud Platform (GCP) con el objetivo de establecer un control de acceso adecuado, seguro y funcional para los recursos que serán utilizados en el despliegue de los sistemas desarrollados. Para ello se realizó lo siguiente:

* **Creación de cuenta de servicio**

Se creó una cuenta de servicio específica denominada Distribuidos-Servicios, la cual será utilizada posteriormente para automatizar despliegues, subir imágenes de contenedores y gestionar recursos de GCP relacionados con microservicios, servicios web y PWA.

* **Asignación de roles con privilegios mínimos necesarios**

A dicha cuenta de servicio se le asignaron los roles de IAM (Identity and Access Management) más adecuados para sus funciones, aplicando el principio de mínimo privilegio. Los roles asignados fueron:

| **Rol asignado** | **Función principal** |
| --- | --- |
| Administrador del repositorio de Artifact Registry para envíos con creaciones | Permite subir imágenes Docker y crear repositorios en Artifact Registry. |
| Administrador de objetos de Storage | Permite crear, leer, actualizar y eliminar objetos dentro de los buckets de Cloud Storage. |
| Administrador de Monitoring | Permite acceder a métricas, crear alertas y configurar paneles de supervisión. |
| Cloud Run Admin | Permite desplegar, configurar y administrar servicios en Cloud Run. |

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### **Configuración de almacenamiento**

Como parte de la preparación del entorno en Google Cloud Platform, se configuraron dos buckets en Cloud Storage, orientados a proporcionar una solución de almacenamiento confiable, organizada y adaptable a los distintos componentes del sistema:

* **Bucket distribuidos-backups**

Este bucket está destinado al almacenamiento de respaldos y recursos internos, incluyendo archivos de base de datos (como SQLite), logs u otros archivos de soporte que los microservicios puedan necesitar conservar o recuperar.

Configuración aplicada:

* Ubicación: us-central1
* Clase de almacenamiento: Nearline. Este es ideal para archivos que se acceden ocasionalmente, optimizando costos sin sacrificar disponibilidad.
* Eliminación no definitiva: Activada, la cual permitirá recuperar archivos eliminados accidentalmente durante un periodo de gracia de 7 días.
* Retención: Predeterminada del sistema (7 días)
* Control de versiones: No habilitado
* Acceso público: Restringido (privado)
* Control de acceso: Uniforme, gestionado mediante IAM
* **Bucket distribuidos-pwa-assets**

Este bucket se utiliza para hospedar archivos estáticos del frontend de la PWA, tales como archivos .html, .css, .js, e imágenes, permitiendo su acceso desde navegadores web.

Configuración aplicada:

* Ubicación: us-central1
* Clase de almacenamiento: Standard. Este es recomendado para contenido accedido frecuentemente por usuarios finales.
* Eliminación no definitiva: Activada para ofrecer protección ante eliminaciones accidentales.
* Retención: Predeterminada del sistema (7 días)
* Control de versiones: No habilitado
* Acceso público: Permitido (se desactivó la prevención de acceso público para poder servir contenido web)
* Control de acceso: Uniforme. Este es gestionado por IAM, con permisos públicos aplicados de forma puntual a los archivos que lo requieren (como index.html)

Esta configuración permite contar con una solución de almacenamiento segura, eficiente y preparada para el despliegue de los servicios y el frontend, adaptándose tanto a las necesidades internas del sistema como a la presentación web orientada al usuario.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

### **Configuración de monitoreo y logging**

Como parte de la preparación del entorno en Google Cloud Platform, se configuraron las herramientas de monitoreo y registro ofrecidas por GCP. Esto permitirá supervisar el comportamiento de los servicios una vez que se encuentren en funcionamiento.

Se creó un entorno de Cloud Monitoring asociado al proyecto, el cual permite consultar métricas, configurar alertas y visualizar el estado general de los recursos. También se generó un panel básico de monitoreo, que servirá como base para agregar gráficas y estadísticas relacionadas con los microservicios que serán desplegados más adelante.

Por otro lado, se dejó habilitado Cloud Logging, que registrará automáticamente los eventos y actividades de los servicios. Esto facilitará la detección de errores y el seguimiento del funcionamiento del sistema una vez que se encuentre en operación.

Interfaz de usuario gráfica, Texto, Aplicación

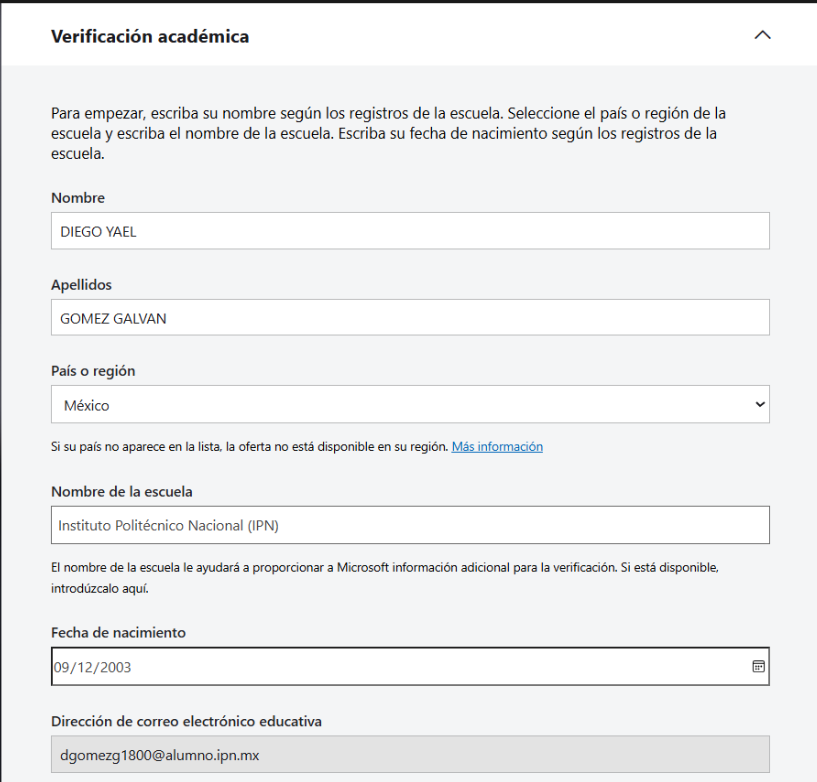
El contenido generado por IA puede ser incorrecto.

## **Microsoft Azure**

Para la configuración básica de la nube Microsoft Azure, realizamos lo siguiente:

### **Creación de cuenta**

Primeramente entramos a la página oficial de Azure donde creamos una cuenta escolar con el correo institucional del ipn, ingresando los datos que se nos solicitaban



Después de haber ingresado los datos, obtuvimos un crédito de 300 dólares durante un año para utilizar en los diferentes servicios que ofrece Azure

### **Crear grupo de recursos**

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.Se creó un grupo de recursos denominado PracticasAzure, el cual sirve como una unidad lógica para agrupar y administrar todos los servicios relacionados con los sistemas distribuidos que serán desplegados en Azure. Este grupo permite centralizar la gestión, monitoreo, permisos y facturación de los recursos, facilitando su organización y mantenimiento. Todos los elementos esenciales del entorno, como el almacenamiento, contenedores, servicios web, monitoreo y despliegue se configuraron dentro de este grupo para mantener una estructura clara y ordenada.

### **Crear cuenta de almacenamiento**

Se creó una cuenta de almacenamiento en Azure dentro del grupo de recursos PracticasAzure, con el objetivo de administrar de forma segura y eficiente los archivos asociados al sistema. Esta cuenta proporciona una solución escalable para guardar datos estructurados y no estructurados, y será utilizada principalmente para dos fines:

* Almacenar respaldos internos y archivos técnicos, como bases de datos SQLite, registros o configuraciones.
* Alojar archivos estáticos del frontend de la PWA, como imágenes, hojas de estilo y scripts JavaScript.

Esta configuración permite mantener separados los datos sensibles y los accesibles públicamente, garantizando así una gestión segura y estructurada del almacenamiento.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### **Configurar contenedores dentro del almacenamiento**

Dentro de la cuenta de almacenamiento, se configuraron dos contenedores con propósitos distintos:

* Contenedor $logs: Este contenedor está destinado a almacenar archivos internos del sistema, como respaldos de bases de datos, archivos de configuración o registros técnicos generados por los microservicios. Se configuró con acceso privado, asegurando que solo los usuarios o servicios autorizados puedan leer o modificar su contenido.
* Contenedor pwa-assets: Este contenedor fue creado para alojar los archivos estáticos del frontend de la Progressive Web App (PWA), como los archivos .html, .css, .js e imágenes. Se configuró con nivel de acceso "Anónimo", lo cual permite que los archivos se sirvan públicamente a través de Internet mediante una URL como:  
  *https://<nombre-de-la-cuenta>.blob.core.windows.net/pwa-assets/index.html*  
  Esta configuración es esencial para permitir que la PWA funcione como sitio web accesible desde cualquier navegador sin necesidad de autenticación.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### **Configurar control de acceso**

Se configuró el control de acceso a nivel del grupo de recursos PracticasAzure, estableciendo los permisos necesarios para poder gestionar los servicios relacionados con el despliegue. Para ello, se utilizó la asignación de roles basados en funciones (RBAC), lo cual permite definir con precisión qué acciones puede realizar cada usuario o servicio.

Al usuario principal, identificado con mi correo institucional, le asigne los siguientes roles:

* Contributor: Permite crear, modificar y eliminar cualquier recurso dentro del grupo, excepto asignar permisos a otros usuarios.
* Storage Blob Data Contributor: Otorga acceso completo de lectura y escritura a los datos almacenados en contenedores de blobs, necesario para administrar los archivos del sistema y los assets de la PWA.

Esta configuración garantiza que el entorno esté listo para operar con seguridad y que los recursos puedan ser gestionados correctamente en futuras fases del proyecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

### **Crear registro de contenedores**

Se creó un registro de contenedor (ACR) dentro del grupo de recursos PracticasAzure, el cual servirá para almacenar imágenes Docker de los microservicios y servicios web que conforman el backend de los sistemas distribuidos. Este registro permite tener un repositorio privado, seguro y completamente integrado con Azure, desde donde se podrán desplegar contenedores directamente hacia servicios como App Service o Azure Kubernetes Service (AKS).

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

### **Crear recurso Application Insights**

Se integró Application Insights al entorno de desarrollo dentro de Azure, como parte del monitoreo esencial para los servicios que serán desplegados más adelante. Esta herramienta permite recopilar métricas, registrar eventos, errores y analizar el rendimiento de las aplicaciones en tiempo real.

Durante la configuración básica se realizaron los siguientes pasos:

* Se creó un recurso de tipo Application Insights en la misma región y grupo de recursos (PracticasAzure).
* Se vinculó con el entorno general de servicios para permitir su integración futura con microservicios o aplicaciones web desplegadas en Azure.
* Se dejó preparado para recibir logs de rendimiento, trazas y excepciones, una vez que los servicios estén en ejecución.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

# **RESULTADOS**

Como parte del proceso de preparación para el despliegue de los sistemas distribuidos, se configuraron distintos servicios dentro de la plataforma Microsoft Azure y Google Cloud Platform. Estos recursos abarcan desde almacenamiento, monitoreo y contenedores, hasta servicios de aplicación y redes de seguridad. A continuación, se muestra una vista general de los recursos creados en el grupo de trabajo correspondiente, los cuales forman la base sobre la que se desplegarán los sistemas desarrollados.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.



Gráfico, Gráfico de barras

El contenido generado por IA puede ser incorrecto.

Todos estos recursos generaron nos servirán de base para realizar próximamente el despliegue de los sistemas realizados.

# **CONCLUSIONES**

Durante esta práctica, aprendí a configurar de forma básica tres de las principales plataformas de nube pública: Google Cloud Platform, Microsoft Azure y Amazon Web Services. Este proceso fue fundamental para sentar las bases del despliegue futuro de mis proyectos desarrollados con arquitecturas de microservicios, servicios web RESTful y Progressive Web Apps (PWA).

Uno de los principales aprendizajes fue comprender cómo cada proveedor estructura sus servicios y recursos, desde el almacenamiento y la seguridad, hasta el monitoreo y la gestión de contenedores. A pesar de sus diferencias, pude identificar elementos comunes como la necesidad de configurar cuentas, crear proyectos o suscripciones, activar APIs, establecer políticas de acceso mínimo y preparar entornos de almacenamiento seguros.

En el caso de Google Cloud Platform, me resultó especialmente útil trabajar con Artifact Registry, Cloud Run y los buckets de almacenamiento para respaldos y assets web. En Azure, comprendí mejor el control de acceso mediante roles (RBAC), la utilidad de los grupos de recursos y la integración con Application Insights. Aunque AWS no se desarrolló a profundidad en este documento, su inclusión me permitió visualizar la necesidad de dominar múltiples entornos para una arquitectura verdaderamente flexible y escalable.

También enfrenté desafíos, donde algunos de los más relevantes fueron configurar correctamente los permisos de acceso sin comprometer la seguridad, entender las distintas clases de almacenamiento y hacer que los archivos estáticos de la PWA estuvieran disponibles públicamente sin exposición innecesaria. Superar estos retos me permitió fortalecer mis conocimientos sobre infraestructura como servicio (IaaS), automatización, monitoreo y despliegue continuo.

En general, esta práctica no solo me permitió consolidar conceptos técnicos, sino que también me dio una visión clara de lo que implica preparar un entorno en la nube real para alojar sistemas distribuidos. Ahora sé que una buena configuración inicial facilita enormemente las fases posteriores del desarrollo, despliegue y mantenimiento.