

Elaborar plan de validación de características mínimas de hardware para el software. GA10-

20501097-AA1-AA2AA3-EV01

Juan Luis Becquet Martínez

Servicio Nacional de Aprendizaje

(2721441)Análisis y desarrollo de software

Christian Leonardo Ospina Hamon

19 de Octubre de 2024

Sumario

Introducción.....	3
1. VPS (Servidor Privado Virtual).....	4
2. Servicios de PaaS (Plataforma como Servicio).....	4
1. Amazon Web Services (AWS).....	5
2. Google Cloud Platform (GCP).....	6
3. Microsoft Azure.....	6
Comparativa General:.....	7
¿Cuál elegir?.....	7
4. Alojamiento Compartido (Shared Hosting).....	8
Ahondemos un poco mas en Heroku.....	9
1. Características Clave de Heroku.....	9
2. Despliegue en Heroku.....	9
3. Escalabilidad.....	9
4. Add-ons.....	10
5. Planes y Precios.....	10
6. Desventajas de Heroku.....	11
Comparemos los servicios de AWS y heroku(los que me parecen mas interesantes)....	12
Conclusión.....	15
Bibliografía.....	16

Introducción.

Las tecnologías en la nube han transformado radicalmente la forma en que las empresas y desarrolladores gestionan, despliegan y escalan sus aplicaciones. Dos de los modelos más utilizados en este ámbito son la **Plataforma como Servicio (PaaS)** y la **Infraestructura como Servicio (IaaS)**. Ambos enfoques permiten la creación y gestión de aplicaciones sin la necesidad de adquirir y mantener hardware físico, pero difieren significativamente en cuanto al control, la flexibilidad y la gestión requerida. Este trabajo busca explorar las ventajas y desventajas de ambos modelos, con el objetivo de proporcionar una visión comparativa clara para ayudar a los desarrolladores y organizaciones a tomar decisiones informadas a la hora de elegir la plataforma más adecuada para sus necesidades.

Para alojar la aplicación web en Linux, hay varias opciones de hosting que se pueden considerar, dependiendo de las necesidades de tráfico, presupuesto y nivel de control sobre la infraestructura. algunas opciones populares:

1. VPS (Servidor Privado Virtual)

Un VPS da control total sobre el servidor, permitiéndolo configurar el sistema operativo (en este caso, Linux), la base de datos PostgreSQL, Flask y cualquier otro servicio. Es una buena opción si tengo conocimientos de administración de servidores (que no es el caso) o si contratar a alguien que lo haga.

- **DigitalOcean**
 - Precios: desde \$5 USD/mes (1GB de RAM, 1 CPU, 25GB SSD)
 - Puedes escalar a medida que aumenten las necesidades de recursos.
 - Tienes control total sobre la instalación y configuración del servidor.
- **Linode**
 - Precios: desde \$5 USD/mes (1GB de RAM, 1 CPU, 25GB SSD)
 - Ofrece características similares a DigitalOcean.
- **Vultr**
 - Precios: desde \$5 USD/mes (1GB de RAM, 1 CPU, 25GB SSD)
 - Variedad de ubicaciones globales para el servidor.

2. Servicios de PaaS (Plataforma como Servicio)

Con esta opción, no hay que preocuparse por la administración del servidor. La plataforma se encarga de la infraestructura, permitiéndolo concentrarse en el desarrollo y despliegue de la aplicación.

- **Heroku**
 - Precios: Plan gratuito disponible (limitado en cuanto a horas y capacidad de base de datos).
 - Planes pagos: desde \$7 USD/mes.
 - PostgreSQL está disponible como un complemento con varios niveles.
- **Render**
 - Precios: desde \$7 USD/mes (para aplicaciones Flask).
 - Bases de datos PostgreSQL desde \$7 USD/mes.

1. Amazon Web Services (AWS)

AWS es uno de los proveedores más grandes de servicios en la nube y ofrece una amplia gama de servicios para alojar aplicaciones Flask y bases de datos PostgreSQL. Aquí algunos detalles de los servicios relevantes:

- **EC2 (Amazon Elastic Compute Cloud):** Es el servicio de instancias virtuales (servidores). Puedes lanzar servidores Linux (Ubuntu, CentOS, etc.) y configurar la aplicación Flask, PostgreSQL y otros servicios. Es muy flexible y permite escoger el tamaño de la instancia según las necesidades.
 - **Precios:**
 - **t2.micro** (instancia más pequeña): Gratis durante 12 meses si se usa el plan gratuito de AWS (1 vCPU, 1GB de RAM).
 - Después del plan gratuito: Aproximadamente \$3.50 USD/mes para una instancia reservada.
 - Las instancias pueden variar en precio dependiendo del tipo (CPU, memoria, etc.).
 - **Escalabilidad:** Puedes aumentar o disminuir la capacidad del servidor (más CPU, más RAM) con facilidad a medida que lo necesites.
- **RDS (Amazon Relational Database Service):** Es un servicio gestionado para bases de datos relacionales, incluido PostgreSQL. RDS se encarga de la administración, actualización, y seguridad de la base de datos, lo que simplifica mucho el mantenimiento.
 - **Precios:**
 - **Instancia db.t3.micro** (PostgreSQL): A partir de \$12 USD/mes.
 - Backup automático y recuperación ante desastres incluidas.
- **S3 (Simple Storage Service):** Para almacenamiento de archivos estáticos como imágenes, archivos PDF o backups. Es útil para almacenar grandes cantidades de datos de manera económica.
 - **Precios:** Depende del uso, pero el costo inicial es de \$0.023 por GB almacenado.
- **Características clave:**
 - **Seguridad:** AWS tiene un robusto sistema de seguridad con políticas de acceso, encriptación, y auditorías.
 - **Escalabilidad:** Automática o manual. Puedes escalar la infraestructura con mayor capacidad o habilitar balanceo de carga.
 - **Región:** se puede seleccionar en qué región geográfica (datacenter) se desea alojar la aplicación para optimizar latencia.

2. Google Cloud Platform (GCP)

GCP es otro proveedor importante que ofrece soluciones en la nube similares a AWS con algunas diferencias en su oferta y precios.

- **Compute Engine:** Es el servicio de máquinas virtuales similar a EC2. Se puede configurar el servidor Linux, instalar Flask, PostgreSQL, y gestionar la infraestructura.
 - **Precios:** Desde \$5-6 USD/mes para instancias pequeñas (f1-micro o e2-micro con 1 vCPU, 0.6 GB de RAM).
 - Hay una oferta de prueba gratuita de 90 días con \$300 de crédito para nuevos usuarios.
- **Cloud SQL para PostgreSQL:** Es una solución de base de datos gestionada que ofrece PostgreSQL, donde GCP se encarga de las actualizaciones, parches de seguridad y backups.
 - **Precios:** Desde \$15 USD/mes para instancias básicas.
 - Cloud SQL facilita la integración con otros servicios de GCP y tiene alta disponibilidad.
- **Cloud Storage:** Equivalente a AWS S3 para almacenar archivos estáticos o grandes cantidades de datos.
 - **Precios:** \$0.02 por GB almacenado, con opciones de clase de almacenamiento (estándar, nearline, etc.) dependiendo de la frecuencia de acceso a los archivos.
- **Características clave:**
 - **Machine Learning y Big Data:** Integraciones avanzadas para análisis de datos y machine learning.
 - **Load balancing y autoscaling:** Herramientas potentes para gestionar el tráfico y escalar automáticamente.
 - **Google Kubernetes Engine (GKE):** Si se desea usar contenedores para desplegar aplicaciones, GCP tiene una integración muy sólida con Kubernetes.

3. Microsoft Azure

Azure, el servicio en la nube de Microsoft, también ofrece una gama completa de soluciones para ejecutar aplicaciones web y bases de datos gestionadas.

- **Azure Virtual Machines:** Máquinas virtuales para ejecutar el servidor Linux, donde se puede instalar Flask y PostgreSQL.
 - **Precios:** Comienzan alrededor de \$4-5 USD/mes para instancias pequeñas (B1S, 1 vCPU, 1 GB RAM).

- Ofrecen un plan gratuito durante 12 meses, con una VM Linux gratuita para usuarios nuevos (750 horas al mes).
- **Azure Database for PostgreSQL:** Servicio gestionado de PostgreSQL que incluye backups automáticos, escalabilidad y actualizaciones automáticas.
 - **Precios:** Desde \$15 USD/mes para una instancia básica.
- **Azure Blob Storage:** Equivalente a S3 y Cloud Storage para el almacenamiento de archivos estáticos y backups.
 - **Precios:** Alrededor de \$0.0184 por GB almacenado.
- **Características clave:**
 - **Integración con servicios de Microsoft:** Especialmente útil si ya se usan servicios como Active Directory, Office 365 o Visual Studio.
 - **Seguridad y Compliance:** Azure tiene políticas robustas de seguridad y cumplimiento normativo que lo hacen popular entre grandes empresas.
 - **Automatización:** Azure ofrece una excelente capacidad de automatización a través de scripts y plantillas (Azure Resource Manager).

Comparativa General:

Característica	AWS	GCP	Azure
Precio inicial	Desde \$3.50/mes (EC2)	Desde \$5/mes (Compute)	Desde \$4/mes (VMs)
Base de datos PostgreSQL	Desde \$12/mes (RDS)	Desde \$15/mes (Cloud SQL)	Desde \$15/mes (Azure DB)
Prueba gratuita	12 meses (EC2, t2.micro)	\$300 crédito/90 días	12 meses (750 horas VM)
Facilidad de uso	Flexible pero requiere gestión	Flexible, buena integración	Buena integración con MS
Escalabilidad	Alta	Alta	Alta
Seguridad	Fuerte	Fuerte	Fuerte

¿Cuál elegir?

- **AWS** es ideal si buscas una solución madura y altamente escalable con gran cantidad de servicios disponibles.
- **GCP** es una buena opción si se prefiere una integración avanzada con Big Data, Machine Learning y se desea probar con créditos gratis.

- **Azure** es adecuado si ya se usan otros servicios de Microsoft o si se planea una integración con plataformas corporativas de Microsoft.

4. Alojamiento Compartido (Shared Hosting)

Es la opción más barata, pero con menos control y recursos. No es tan flexible como VPS o PaaS, pero algunos proveedores soportan tecnologías como Flask.

- **A2 Hosting**
 - Precios: desde \$2.99 USD/mes.
 - Soporte para Python (Flask) y PostgreSQL.
- **Hostinger**
 - Precios: desde \$1.99 USD/mes.
 - No siempre incluye soporte para PostgreSQL de manera sencilla, pero puedes usar otros servicios externos para la base de datos.

Ahondemos un poco mas en Heroku

Heroku es una plataforma como servicio (PaaS) que facilita el desarrollo, despliegue y escalabilidad de aplicaciones web sin necesidad de gestionar infraestructura compleja. Es ideal para desarrolladores que buscan una forma rápida y eficiente de lanzar aplicaciones sin tener que preocuparse por la administración de servidores.

1. Características Clave de Heroku

- **Fácil de Usar:** Heroku simplifica enormemente el proceso de despliegue. Lo que se necesita es preparar la aplicación (por ejemplo, con Flask), conectarla a un repositorio (como GitHub), y luego usar comandos simples para desplegarla.
- **Compatibilidad con Lenguajes:** Soporta múltiples lenguajes, incluyendo **Python** (ideal para Flask), **Node.js**, **Ruby**, **Java**, **PHP**, entre otros. Esto lo hace flexible si en el futuro si se necesita agregar microservicios en otros lenguajes.
- **Add-ons:** Heroku tiene una gran cantidad de **add-ons** que permiten integrar servicios adicionales fácilmente, como bases de datos, herramientas de monitoreo, almacenamiento en caché, gestión de colas, entre otros.
- **Escalabilidad Automática:** Se puedes escalar la aplicación rápidamente simplemente ajustando el número de **dynos** (unidades de proceso que ejecutan la aplicación). Esto es útil si el tráfico aumenta de manera repentina.
- **Gestión Automática de la Infraestructura:** Heroku gestiona automáticamente todo lo relacionado con la infraestructura subyacente, como la actualización del sistema operativo, la aplicación de parches de seguridad y la asignación de recursos.

2. Despliegue en Heroku

- **Deploy desde GitHub:** Puede conectar directamente el repositorio de **GitHub** a Heroku, permitiendo despliegues automáticos cada vez que haces un cambio en el código.
- **Deploy desde Git:** También puede hacer el despliegue directamente desde el sistema local con `git push heroku main`.
- **Pipeline de CI/CD:** Heroku ofrece una integración directa con sistemas de CI/CD (Continuous Integration/Continuous Deployment), permitiendo hacer pruebas y desplegar automáticamente a producción cuando las pruebas son exitosas.

3. Escalabilidad

- Heroku facilita mucho el proceso de escalabilidad. A medida que el tráfico de la aplicación crezca, puedes aumentar el número de **dynos** fácilmente con un comando.
- Puedes escalar tanto **horizontal** como **verticalmente**:

- **Escalabilidad horizontal:** Aumentar el número de dynos para gestionar más peticiones.
- **Escalabilidad vertical:** Aumentar los recursos asignados a cada dyno, como la cantidad de RAM o CPU.

4. Add-ons

Heroku tiene un **Marketplace** con más de 200 add-ons para mejorar la aplicación. Algunos de los más útiles para el proyecto podrían ser:

- **Heroku Postgres:** Un servicio de bases de datos gestionadas PostgreSQL. Tiene un plan gratuito que permite almacenar hasta 1,000 filas, con opciones pagas para bases de datos más grandes.
- **Redis:** Para implementar caching en la aplicación y mejorar el rendimiento.
- **Papertrail:** Para almacenar y consultar logs de la aplicación, útil para depurar problemas.
- **SendGrid:** Para enviar emails desde la aplicación, por ejemplo, para notificaciones o restablecimiento de contraseñas.
- **New Relic:** Para monitoreo de rendimiento y análisis de la aplicación.

5. Planes y Precios

Heroku ofrece varios planes que se ajustan a diferentes necesidades. Aquí están los más relevantes:

- **Plan Free (Gratuito):**
 - Este plan permite desplegar aplicaciones en servidores compartidos. Las aplicaciones entran en modo de "hibernación" después de 30 minutos de inactividad
 - Limitado a 550 horas de ejecución de dynos al mes, lo que equivale aproximadamente a 22 días de ejecución continua.
 - Útil para pruebas o aplicaciones con bajo tráfico, pero no para producción si esperas que los usuarios accedan a la aplicación en cualquier momento.
 - Heroku Postgres: 1,000 filas de datos de PostgreSQL gratis.
- **Plan Hobby (\$7 USD/mes por dyno):**
 - Es más estable que el plan gratuito y no entra en modo hibernación.
 - Ideal para pequeñas aplicaciones en producción con tráfico bajo o medio.
 - Heroku Postgres: Almacena más datos (en la versión paga de PostgreSQL).
- **Plan Standard y superiores (a partir de \$25 USD/mes por dyno):**

- Estos planes son adecuados para aplicaciones que necesitan más rendimiento y mayor capacidad de escalado.
- Incluyen dynos más potentes, con mejor rendimiento y más capacidad de procesamiento.
- Ideal para aplicaciones con tráfico moderado o alto.
- **Add-ons Pagos:** Los precios de los add-ons también varían. Por ejemplo, si se opta por una base de datos PostgreSQL gestionada de mayor capacidad, los precios comienzan en **\$9 USD/mes** y suben dependiendo del tamaño y el rendimiento necesario.

6. Desventajas de Heroku

- **Costo:** A medida que la aplicación crece, Heroku puede volverse más caro que opciones como un VPS. Aunque es muy conveniente, si tu aplicación consume muchos recursos (por ejemplo, con muchas consultas a la base de datos), los costos pueden escalar rápidamente.
- **Limitaciones de Personalización:** Aunque Heroku es muy flexible, no tienes control total sobre la infraestructura subyacente como tendrías con un VPS o un servidor dedicado.
- **Dynos durmientes (en plan gratuito):** Las aplicaciones en el plan gratuito se suspenden cuando no se usan durante un tiempo, lo que significa que los usuarios podrían experimentar un retraso cuando intenten acceder por primera vez tras un periodo de inactividad.

Comparemos los servicios de AWS y heroku(los que me parecen mas interesantes)

Característica	PaaS (Plataforma como Servicio)	IaaS (Infraestructura como Servicio)
Definición	Plataforma completa que gestiona el entorno de desarrollo, despliegue y escalado de aplicaciones. Ejemplos: Heroku, Google App Engine, AWS Elastic Beanstalk.	Provisión de recursos de infraestructura como servidores, redes y almacenamiento. Ejemplos: AWS EC2, Google Cloud Compute, Azure VMs.
Control	Control limitado sobre la infraestructura. Los proveedores gestionan el sistema operativo, actualizaciones, y configuraciones.	Control total sobre el sistema operativo, red, almacenamiento y configuraciones de los servidores.
Facilidad de uso	Muy fácil de usar. Los desarrolladores se enfocan solo en el código y el despliegue. No requieren conocimientos profundos de la infraestructura.	Requiere más conocimientos técnicos para gestionar, configurar y mantener la infraestructura.
Flexibilidad	Limitada en cuanto a personalización del entorno y configuración avanzada. Ideal para aplicaciones estándar.	Alta flexibilidad y personalización. Puedes configurar el entorno exactamente como desees.
Escalabilidad	Escalabilidad automática o gestionada fácilmente por el proveedor. Menos trabajo manual para aumentar recursos.	La escalabilidad es manual y requiere intervención del usuario para gestionar la capacidad.
Despliegue	Despliegue rápido y sencillo. No requiere gestión de servidores, solo especificar los recursos necesarios.	El despliegue es más complejo, ya que implica configurar servidores, redes y almacenamiento antes de desplegar la aplicación.
Costos iniciales	Inicialmente más económico. No hay necesidad de comprar	Mayor costo inicial en comparación, ya que pagas por

	hardware o mantener servidores. Planes gratuitos disponibles en algunas plataformas.	la infraestructura directamente (CPU, RAM, almacenamiento). Los costos dependen del uso exacto de los recursos.
Costo total	Puede volverse costoso al escalar si no se optimiza bien el uso de recursos, especialmente si se requieren muchos add-ons o servicios adicionales.	Generalmente más económico a gran escala, ya que puedes optimizar el uso de recursos y pagar solo por lo que usas, aunque requiere más gestión.
Mantenimiento	Sin preocupaciones por el mantenimiento del sistema operativo, parches de seguridad o actualizaciones. Todo esto es gestionado por el proveedor.	El mantenimiento es responsabilidad del usuario, incluyendo parches, actualizaciones y seguridad del sistema operativo.
Seguridad	Seguridad gestionada por el proveedor en términos de protección del sistema y la infraestructura, aunque el usuario debe gestionar la seguridad de la aplicación.	Mayor control sobre la seguridad, pero también mayor responsabilidad del usuario para configurar firewalls, políticas de acceso y otras protecciones.
Casos de uso	Ideal para startups, proyectos pequeños o medianos, y aplicaciones web con configuraciones estándar que requieren rápido despliegue.	Adecuado para aplicaciones empresariales grandes, con necesidades personalizadas, configuraciones complejas o aquellas que requieren alta flexibilidad.
Dependencia del proveedor	Alta. Los desarrolladores dependen de la infraestructura y características específicas del proveedor. Migrar a otra plataforma puede ser complejo.	Menor dependencia. Aunque estés usando un proveedor de IaaS, puedes mover tus recursos a otra infraestructura con menos complicaciones.
Soporte para tecnologías emergentes	Los proveedores PaaS suelen actualizar con soporte para nuevas tecnologías rápidamente. Sin embargo, si necesitas algo específico que	Puedes implementar cualquier tecnología o software emergente siempre que puedas configurarlo en los servidores.

	no soportan, puede ser limitante.	
--	-----------------------------------	--

Conclusión

- **PaaS** es ideal si se busca simplicidad, despliegue rápido y no quieres preocupaciones por la gestión de la infraestructura. Es perfecto para startups, proyectos pequeños y medianos, o cuando la velocidad de desarrollo es clave.
- **IaaS** es la opción preferida si se necesita más control sobre la infraestructura, personalización y se está dispuesto a gestionar configuraciones y mantenimiento. Es mejor para proyectos más grandes y complejos que requieren control detallado sobre los recursos.

Al parecer luego de toda la investigación realizada, la mejor decisión y opción para mí es un servidor PaaS, en los cuales encaja el servicio de Heroku, para prácticas iniciales podría funcionar tal servicio, cuando crezcan los clientes se puede realizar una migración completa a otro servicio IaaS que pertenece a AWS.

Bibliografia

Armbrust, M., et al. (2010). "A view of cloud computing." *Communications of the ACM*, 53(4), 50-58.

Hussain, F., et al. (2013). "A framework for cloud platform selection based on provider's ranking." *Journal of Cloud Computing*, 2(1), 1-17.

Buyya, R., et al. (2011). *Cloud Computing: Principles and Paradigms*. Wiley.