

UNIVERSIDAD AMERICANA
FACULTAD DE INGENIERÍA Y ARQUITECTURA



Algoritmo y estructura de datos

Nombre:

Dosson Arnoldo Luques Narvaez.
Anthony King Flores Garcia.

Docente:

SILVIA GIGDALIA TICAY LÓPEZ

Fecha: 17/03/2025

Almacenamiento en la nube con python

Python ha demostrado ser una herramienta fundamental para interactuar con servicios de almacenamiento en la nube, ofreciendo bibliotecas especializadas y soluciones multiplataforma, Python facilita la integración con servicios de almacenamiento en la nube gracias a una amplia variedad de bibliotecas y SDKs oficiales.

Entre los servicios más usados se encuentra Amazon S3, Google Cloud Storage, Azure Blob Storage, Dropbox API y Google Drive API, con Python los usuarios podrán hacer tareas como la carga y descarga de archivos, gestiones de permisos, sincronización de datos en la nube y automatización de respaldos mediante scripts o aplicaciones más complejas. Las principales bibliotecas utilizadas incluyen boto3 para AWS, google-cloud-storage para Google Cloud, azure-storage-blob para Azure y dropbox para Dropbox. Estas herramientas permiten implementar soluciones eficientes para el almacenamiento y recuperación de datos, optimizando costos y mejorando la accesibilidad.

El almacenamiento en la nube con Python se aplica en la automatización de respaldos personales y empresariales, el almacenamiento de vídeos e imágenes y videos en plataformas web, la gestión de documentos compartidos y la integración con aplicaciones móviles y de escritorio.

También podemos desarrollar nuestro propio almacenamiento en la nube con Python sin depender de servicios externos como los anteriormente mencionados, sin embargo este enfoque lo que busca crear una infraestructura necesaria y básica para almacenar, gestionar y acceder a distintos archivos de manera remota, para esto se necesitarán componentes claves como un servidor de almacenamiento el cual es posible usar Flask o FastAPI para crear una API que permita subir y descargar archivos. Con el mismo Flask y HTML/CSS para crear un panel web y administrar los archivos subidos y un cliente en Python para que interactúe con el servidor con request o socket.

Limitaciones y desafíos de crear nuestro propio almacenamiento en la nube con Python

- **Disponibilidad:** Sin una infraestructura distribuida, si el servidor falla, los archivos no estarán accesibles.
- **Escalabilidad:** A medida que aumente el volumen de datos, será necesario implementar almacenamiento distribuido.
- **Seguridad:** Se debe incluir autenticación, cifrado y permisos para evitar accesos no autorizados.

Este enfoque puede ser útil para proyectos personales o de pequeña escala, pero para aplicaciones críticas se recomienda usar servicios especializados.