
Aplicación web de facturación e infraestructura

201801064 – Kevin Eduardo Ruiz Alvarez

Resumen

El presente proyecto trata sobre la venta, facturación y mantenimiento de infraestructura informática organizada en tipos, categorías, recursos, entre otros, Implementando un API que brinde servicios utilizando el protocolo HTTP. La estructura básica de la tecnología de la nube consiste en crear configuraciones de infraestructura que agrupan recursos necesarios para que una empresa pueda construir las arquitecturas de despliegue de aplicaciones que requiera.

Además esta aplicación implementa diferentes tecnologías y herramientas para hacer más eficaz su uso y creación, entre estas esta el uso del lenguaje de metadatos XML, el uso del entorno de trabajo (framework web) Django que nos permita el desarrollo de un sitio web seguro para el consumo de nuestro servidor, todo esto basado en el lenguaje de programación Python y bajo el concepto de programación orientado a objetos (POO) así como el uso de base de datos que en este caso se trabajara bajo el concepto de los archivos XML no efímeros.

Palabras clave

HTTP, API, nube, arquitecturas, entorno

Abstract

This project deals with the sale, billing and maintenance of computer infrastructure organized in types, categories, resources, among others, implementing an API that provides services using the HTTP protocol. The basic structure of cloud technology is to create infrastructure configurations that pool the resources needed for a business to build the application deployment architectures it requires.

In addition, this application implements different technologies and tools to make its use and creation more efficient, among these is the use of the XML metadata language, the use of the Django work environment (web framework) that allows us to develop a secure website for the consumption of our server, all this based on the Python programming language and under the concept of object-oriented programming (OOP) as well as the use of a database that in this case will work under the concept of non-ephemeral XML files.

Keywords

HTTP, API, cloud, architecture, framework

Introducción

La empresa Tecnologías chapinas, S.A está desarrollando una herramienta que sea capaz de facturar detalladamente los servicios de infraestructura de nube que aprovisiona a sus clientes. La estructura de la tecnología de nube desarrollada por tecnologías chapinas consiste en crear configuraciones de infraestructura que agrupan recursos necesarios para que una empresa pueda construir las arquitecturas de despliegue de aplicaciones que requiera. La aplicación web a desarrollar implementa una API que brinda servicios utilizando el protocolo HTTP a través del concepto de programación orientada a objetos (POO) y el uso de bases de datos.

La aplicación web se implementa a través del lenguaje de programación Python, el framework Django para el consumo del servidor, además del microframework flask para la construcción del servidor y el uso del lenguaje de metadatos XML tanto para la administración de los datos como para el uso de base de datos. Se profundizará en estas herramientas y tecnologías dentro de este reporte

Desarrollo del tema

La estructura de la tecnología de nube desarrollada por la empresa tecnologías Chapinas S.A consiste en crear configuraciones, categorías y recursos para que una empresa pueda construir arquitecturas de despliegue para sus aplicaciones, cada uno de estos pilares dentro de la aplicación web sustenta una información valiosa dentro de esta, la organización de cada pila dentro de la aplicación web está dada por:

1. Categoría – Una categoría además de agrupar una lista de configuraciones, tiene un nombre, un id, una descripción y una carga de trabajo
2. Configuración – Una configuración agrupa una lista de recursos para un mejor proceso de administración dentro de la aplicación web, además contiene un id, nombre, descripción para su uso dentro de aplicación.
3. Recurso – un recurso es el pilar central dentro de aplicación web ya que estos son integraciones para brindar software o hardware a la empresa que requiera de estos. Cada recurso que se puede incluir en una configuración posee una dimensionalidad, el costo por hora del recurso, un nombre, tipo y la métrica.
4. Instancia – Representan el aprovisionamiento de una configuración que el usuario puede seleccionar en alguna de las categorías que brinda la empresa.

Actualmente, los clientes de Tecnologías Chapinas, S.A. se registran ingresando su nombre, su NIT, su dirección física y su dirección de correo electrónico, una vez registrados reciben en su correo electrónico un usuario y una clave con la cual pueden ingresar a una consola donde pueden crear, modificar y eliminar instancias. Una vez que el usuario ha aprovisionado las instancias que va a requerir, entonces, cada vez que encienda su instancia, ésta empezará a calcular el tiempo de uso de cada recurso involucrado, y mensualmente, deberá facturar según las tarifas vigentes por cada recurso.

a. Manejo del lenguaje de etiquetado extensible

El uso de esta herramienta para el manejo de datos de una mejor manera comúnmente se llama XML, que

son las siglas de *Extensible markup language*. Estos archivos se componen de etiquetas que nos aportan datos en información que queremos procesar. Estas etiquetas pueden estar de forma individual o anidadas.

Este lenguaje de marcado nació como resultado de la optimización del Lenguaje SGML, un formato anterior que se utilizaba con el mismo propósito. El uso de este ocurre mas que todo porque los datos están almacenados en formato de texto simple, lo que permite almacenar y compartir la información entre diferentes sistemas o aplicaciones de manera más rápida y sencilla, además que estos pueden ser leídos por diferentes plataformas aunque estas sean incompatibles entre sí.

La estructura de un archivo XML es bastante peculiar ya que los elementos deben seguir una estructura de “árbol” (estrictamente jerárquica). Además de que los elementos deben estar correctamente anidados y no se pueden superponer entre ellos. En un documento XML solo puede haber un elemento raíz, en el que estén contenidos los demás.

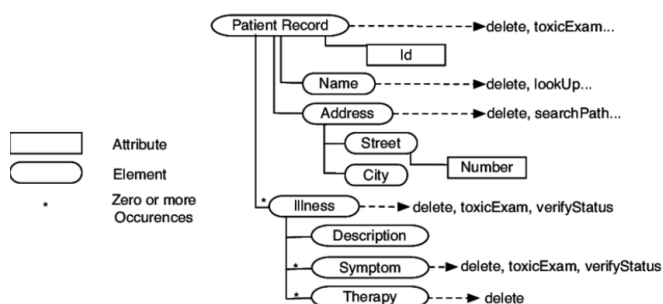


Figura 1. Forma gráfica de un archivo XML.

Fuente: elaboración propia.

b. Implementación de una API

El término API es una abreviatura de Application Programming Interfaces, que en español significa

interfaz de programación de aplicaciones. Se trata de un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones, permitiendo la comunicación entre dos aplicaciones de software a través de un conjunto de reglas.

Así pues, podemos hablar de una API como una especificación formal que establece cómo un módulo de un software se comunica o interactúa con otro para cumplir una o muchas funciones. Todo dependiendo de las aplicaciones que las vayan a utilizar, y de los permisos que les dé el propietario de la API a los desarrolladores de terceros.

¿Cómo funcionan?

La arquitectura de las API suele explicarse en términos de cliente y servidor. La aplicación que envía la solicitud se llama cliente, y la que envía la respuesta se llama servidor. En el ejemplo del tiempo, la base de datos meteorológicos del instituto es el servidor y la aplicación móvil es el cliente.

API de REST

Estas son las API más populares y flexibles que se encuentran en la web actualmente. El cliente envía las solicitudes al servidor como datos. El servidor utiliza esta entrada del cliente para iniciar funciones internas y devuelve los datos de salida al cliente. Veamos la API de REST con más detalle a continuación.



Figura 2. explicación gráfica de una API.

Fuente: www.astera.com.

c. Protocolo HTTP

HTTP, de sus siglas en inglés: "Hypertext Transfer Protocol", es el nombre de un protocolo el cual nos permite realizar una petición de datos y recursos, como pueden ser documentos HTML. Es la base de cualquier intercambio de datos en la Web, y un protocolo de estructura cliente-servidor, esto quiere decir que una petición de datos es iniciada por el elemento que recibirá los datos (el cliente), normalmente un navegador Web. Así, una página web completa resulta de la unión de distintos subdocumentos recibidos, como, por ejemplo: un documento que especifique el estilo de maquetación de la página web (CSS), el texto, las imágenes, vídeos, scripts, etc...

d. Django y flask

Tanto Django como flask son marcos de trabajo (Frameworks) para el desarrollo web. Django es un framework de aplicaciones web gratuito y de código abierto (open source) escrito en Python. Un framework web es un conjunto de componentes que te ayudan a desarrollar sitios web más fácil y rápidamente. Flask es un "micro" Framework escrito en Python y concebido para facilitar el desarrollo de Aplicaciones Web bajo el patrón MVC.

e. Lenguaje de programación Python, POO y diagrama de clases

Python es un Lenguaje de alto nivel, orientado a objetos con una semántica dinámica integrada, principalmente para el desarrollo web y aplicaciones informáticas.

Este lenguaje también es interpretado, esto quiere decir que no se debe compilar el código antes de su ejecución. Python dispone de un intérprete por la línea de comandos en que se puede introducir

sentencias. Cada sentencia se ejecuta y produce un resultado visible.

Python permite varios paradigmas de programación, incluyendo la programación Orientada a objetos (POO). La POO es una manera de estructurar el código que le hace especialmente efectivo organizado y reutilizando código, aunque su naturaleza abstracta hace que no sea muy intuitivo cuando se empieza.

En general, los objetos se pueden considerar como tipos de datos con características propias que también pueden tener funcionalidades propias. Los objetos se definen usando Clases y las variables que se definen en ellas son propiedades comunes de ese objeto.

En este Sistema se utilizaron un total de 6 clases para los modelos de datos del proyecto. y ya en el servidor se trabajó las rutas de y el manejo de estas para la administración de los datos.

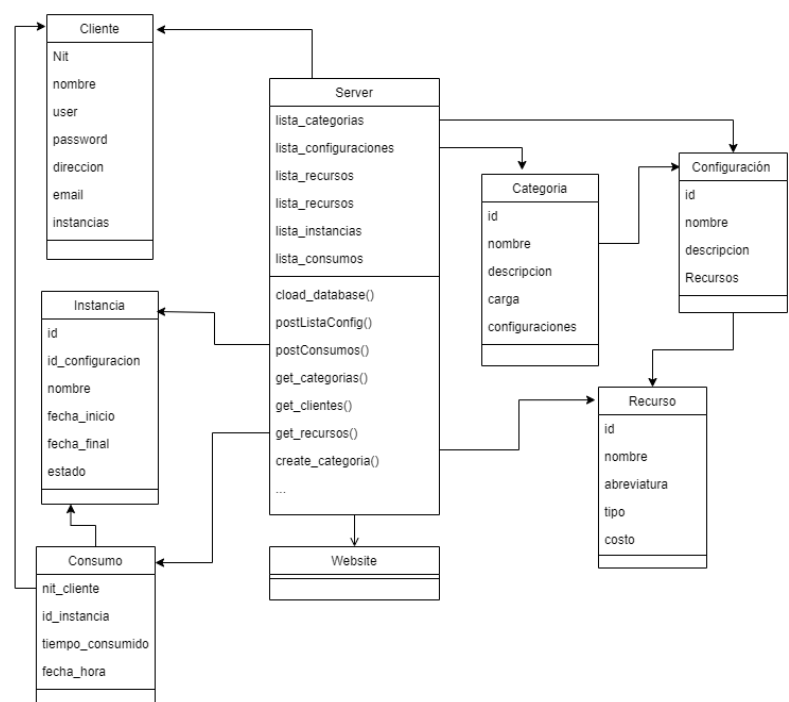


Figura3. Diagrama de clases del proyecto

Fuente: elaboración propia.

Las clases creadas para el proyecto se definieron mediante este diagrama, ya que acá se puede comprender de una mejor manera como esta estructurado el proyecto. La clase principal del sistema como se puede visualizar en el diagrama UML es la clase de server, ya que esta realiza la mayoría de las operaciones que requiere el sistema y conexión con otras clases.

Conclusiones

La implementación de las APIs a las aplicaciones web actualmente son una de las tecnologías mas usadas, a parte que ayudan en la comunicación y administración de estas. Además del uso de los lenguajes de metadatos permiten la facilidad de la administración de los datos.

Referencias bibliográficas

Jacobson, D., Brail, G. & Woods, D. (2012). *Apis: A Strategy Guide: Creating Channels with Application Programming Interfaces* (Illustrated). O'Reilly Media.

Peña, L. (2021, January 31). *PYTHON 3: Parte III - Programación Orientada a Objetos (Aprende Python 3 Desde Cero y Fácilmente)* (Spanish Edition). Independently published.