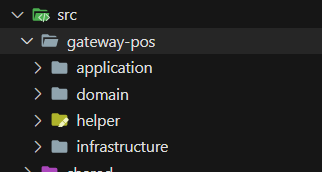
**Cuestionario**

2.- Se utilizo el patrón de arquitectura Hexagonal.



Para la solución de este requerimiento se utilizó el patrón middleware para la gestión de peticiones que utiliza una lista de funciones que actúan como pipe en la entrada y la salida de cada endpoint.

La solución se basó en el principio SOLID donde cada componente se constituye en:

* Principio de responsabilidad única ( cada clase tiene una responsabilidad única)
* Principio de abierto cerrado ( principalmente en el uso de interfaces nos permite extender la capa de infraestructura)
* Principio de sustitución de liskov: el cual actualmente lo estamos empleados empleando en las clase de la capa de infraestructura de un interfaz siendo cada interfaz única para cada clase)
* Principio de inversión de dependencias: el cual lo estamos utilizando al permitir en el constructor definir variables abstractas como interfaces, al utilizar la clase se pueda instancia de cualquier clase que implemente dicha interfaz.