Santiago Ramirez Arenas

Recto Técnico: Training League

ESTACIÓN ESPACIAL SOFKA

¿Quién no ha pensado alguna vez en cómo sería la experiencia de viajar al [espacio](https://www.muyinteresante.es/ciencia/fotos/la-foto-del-dia-del-espacio-2017)?

En este reto vamos a desarrollar un proyecto aplicando conceptos claves de programación orientada a objetos. Con respecto a una lectura se debe descubrir los tipos de naves espaciales y sus características, se debe tener un método que me permita crear y clasiﬁcar estas naves espaciales y un mecanismo para hacer consultas simples y avanzadas. El siguiente es el artículo que se deben basar para la investigación:

[https://moaramore.com/2016/05/14/clasiﬁcacion-de-las-naves-espaciales](https://moaramore.com/2016/05/14/clasificacion-de-las-naves-espaciales) Teniendo en cuenta el contexto anterior, deberás realizar lo siguiente:

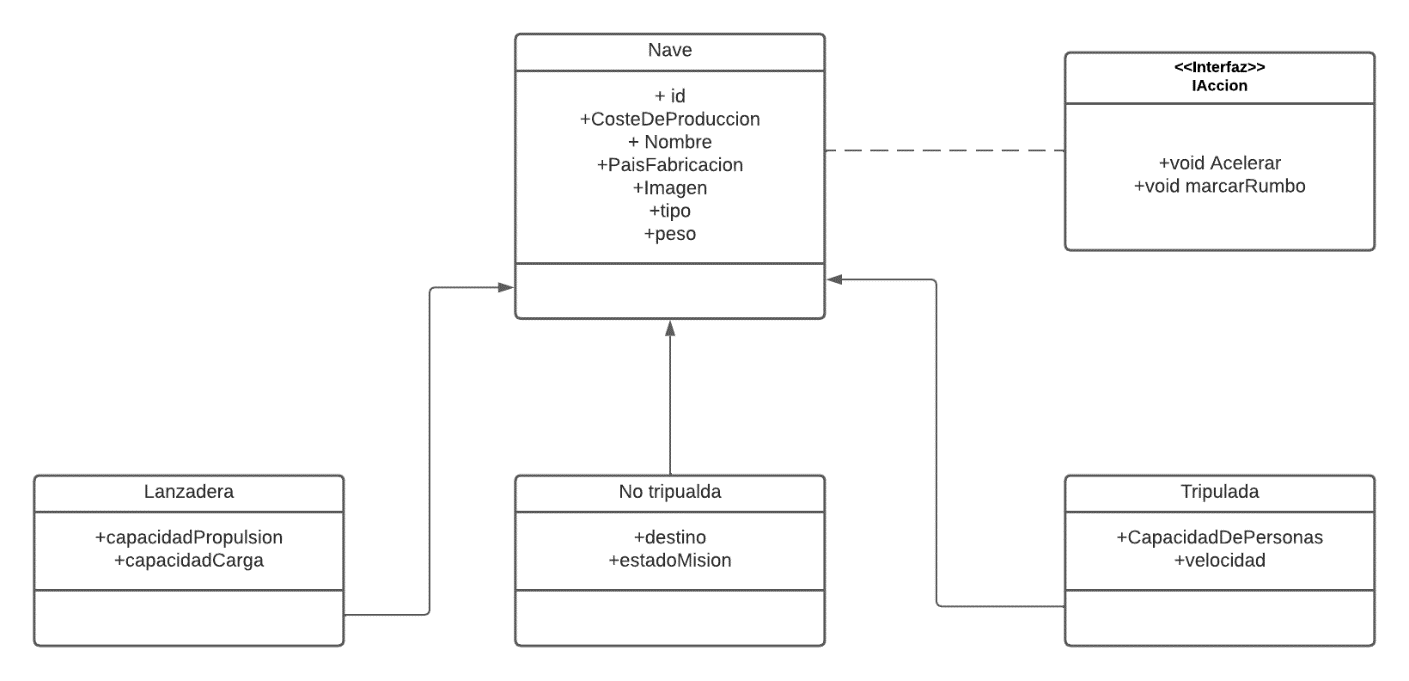
1. Realizar una extracción de atributos genéricos abstractos e interfaces, que puedas identiﬁcar en el artículo. Crear las interfaces y clases abstractas que creas convenientes según la lectura.
2. Crear al menos dos comportamientos abstractos de una nave.
3. Crear al menos cuatro tipos de naves diferentes con diferentes comportamientos, si es posible hacer una sobrecarga y una sobreescritura de métodos.
4. Crear una clase principal que permita crear las naves de forma tal este bajo una condición booleana o por medio de una regla estipulada dentro de un switch-case o if-else.
5. El programa debe tener una ejecución inﬁnita, hasta que el usuario decida terminar el programa en cualquier momento.
6. Con todo lo anterior, se debe crear un programa tipo inventario de mínimo 3 naves de cada posible nave, es decir, sí su análisis dio para crear 3 tipos de naves, entonces usted deberá tener en inventario, cómo mínimo, 9 naves.
7. El sistema deberá contar con los métodos suficientes que me permita buscar una o varias naves según las características de cada nave.

Al ﬁnal se debe entregar un proyecto que contenga una clase principal para poder agregar las naves según sus características y otro método que tenga como argumento un objeto que pueda permitirme ﬁltrar ya sea simple o avanzado.

**SOLUCIÓN**

La arquitectura de una API REST con operaciones CRUD es esencial para la creación de servicios de BackEnd robustos y escalables. En este trabajo se documentará el diseño y desarrollo de una arquitectura de API REST que permita la implementación de las operaciones básicas de una aplicación web para la administración de naves espaciales

**Diagrama de clases**



**Arquitectura API REST con operación CRUD para servicios BackEnd**

