**PERFIL FULLSTACK ALMUNDO**

El objetivo de este ejercicio es representar el trabajo típico de un perfil fullstack.

Sentite libre de agregar/modificar todo lo que consideres necesario para sumarle valor al proyecto, sin perjudicar los tiempos de entrega y la idea central del ejercicio.

Mucha suerte!!

**Ejercicio 1 – API Rest NodeJs:**

Este punto consiste en armar un API Rest en Node.js (para ahorrar tiempo, como base puedes utilizar un stack de desarrollo con el que estés familiarizado). Es necesario crear un método que retorne una lista de hoteles. El mismo será utilizado en el ejercicio 2.

**Extras/Plus:**

Los plus son adicionales que suman valor a tu ejercicio, aunque no son obligatorios.

\*Uso de patrones de diseño y best practices.

\*Uso de **ES6o TYPESCRIPT**

\*Uso de herramientas como webpack, gulp o algun gestor de tareas.

Modelo de respuesta:

{

"hotels" : [

{

"name" : "Hotel Emperador",

"stars" : "3",

"price" : "1596,

},

{

"name" : "Petit Palace San Bernardo",

"stars" : "4",

"price" : "2145",

},

{

"name" : "Hotel Nuevo Boston",

"stars" : "2",

"price" : "861",

}

]

}

**Ejercicio 2 - Frontend:**

Maquetar una página de resultado de hoteles, solo los clusters de hoteles y filtros (ver adjunto).

Consumir la Api desarrollada en el ejercicio anterior, con angular.js 1.x o 2.x, para listar los hoteles dinámicamente.

Tener en cuenta:

- Grilla a elección para el maquetado.

- Uso de gestor de tareas para el tratamiento de estáticos (minificación, ofuscación, etc)

- Uso de patrones de diseño y best practices (componentes, filtros, factories, etc).

Ante alguna duda puede consultar esta [página](https://www.google.com/)