



CAMPEONATO DE ARAGÓN DE FORMACIÓN PROFESIONAL

17 DISEÑO WEB

Participante: _____
Fecha: *19 de Mayo de 2016*



Participante: _____

Fecha: 19 de Mayo de 2016

Descripción general de la prueba

Desarrollar una aplicación web que muestre datos sobre distintas cervezas.

La arquitectura de la aplicación consistirá en un pequeño desarrollo en servidor (API REST) y otro en cliente.

La parte de servidor se debe realizar mediante **javascript, php o python**. Es opcional el uso de algún tipo de base de datos. Se puede utilizar cualquier tipo de framework.

La parte cliente se debe realizar con html, css (o algún lenguaje de preprocesado tipo LESS o SASS) y JavaScript. Se puede utilizar cualquier tipo de framework.

Estrategias

El desarrollo de cada parte puede ser independiente: por ejemplo puedes hacer inicialmente el desarrollo en cliente simulando la capa API del servidor o al revés, realizar la API en el servidor simulando las consultas del cliente.

También puedes optar por trabajar con el JSON que proporcione y no hacer consultas AJAX a la API. Es importante que valores cada parte del enunciado con la puntuación del mismo y optes por tu mejor estrategia para su desarrollo.

Para poder testear el API del servidor (métodos http: POST, PUT Y DELETE) será necesario utilizar alguna herramienta tipo POSTMAN, tests o tener la parte cliente completada.

Para testear el control de versiones se utilizarán los comandos de git:

git log

git diff <commit1> <commit2>

Los apartados de la prueba que puntúan 0,5p tendrán que estar bien en su totalidad para puntuar.

Parte 1 - BACKEND (3,5P)

- Obtener un listado con todas las cervezas **(0,5p)**
- Obtener los datos de una cerveza en función de su identificador **(0,5p)**
- Actualización de los datos de una cerveza en función de su identificador **(0,5p)**
- Borrado de una cerveza en función de su identificador **(0,5p)**
- Dar de alta una nueva cerveza **(0,5p)**
- Mostrar error 404 al consultar una cerveza con un identificador erróneo. **(0,5p)**
- Se cumple la arquitectura REST en la API **(0,5p)**:
 - resources, endpoints y métodos http
 - versión de la API dentro de la url

Participante: _____

Fecha: 19 de Mayo de 2016

Parte 2 - FRONTEND (5,5P)

- Mostrar una **lista** con el nombre, graduación y precio de cada cerveza, con un botón de edición y otro de borrado. **(0,5p)**
- Implementar el botón de **edición**. Al pulsar se deberá mostrar la ficha de la cerveza: un formulario con todos sus datos. **(0,5p)**
- Implementar el botón de **borrado**. Al pulsar el botón deberá desaparecer la cerveza de la lista. **(0,5p)**
- El sistema solicita **confirmación** antes del **borrado**. **(0,5p)**
- Implementar **modificación** de los datos de una cerveza desde su ficha. **(0,5p)**
- Se podrá **dar de alta** una nueva cerveza. **(0,5p)**
- Se valorará un **diseño creativo***. Debido a la limitación en tiempo, se recomienda el uso de algún framework de frontend tipo Bootstrap o Foundation. **(1p)**
- Se valorará la optimización del código: **(0,5p)**
 - html, css y js minificado
 - Numero reducido de peticiones http (scripts, ficheros css o imágenes)
- Se valorará el uso de alguna **herramienta de automatización** tipo Gulp, Grunt, Browserify o Webpack. **(0,5p)**
- Uso de AJAX para las consultas a la API en vez de simularlas con el fichero JSON que se proporciona. **(0,5p)**

** Debido a la subjetividad de este criterio se valorará entre 0 y 1p en función de los criterios de los jueces en función al comparar la apariencia final de los proyectos presentados.*

PARTE 3 - CONTROL DE VERSIONES (1P)

Control de versiones

- Se han realizado al menos 5 commits. Cada commit tiene un mensaje adecuado y se observa que los ficheros que se modifican corresponden al mensaje del commit. **(0,5p)**
- Fichero README.md realizado en markdown con todos o algunos de los siguientes datos (en función del proyecto): **(0,5p)**
 - Tecnologías utilizadas
 - Acceso a la aplicación.
 - Cómo desplegar la aplicación o cómo generar la versión de producción (en caso de usar alguna herramienta de automatización de tareas)