



IIC3103 - Taller de Integración

Departamento Ciencia de la Computación
Escuela de Ingeniería
Pontificia Universidad Católica

Enunciado Tarea 1

Objetivo

El objetivo de esta tarea es desplegar un sitio web que se conecte a una API Rest para obtener información y mostrarla al público.

Trabajo a realizar

Se debe construir un sitio web que muestre información sobre las películas y personajes de Star Wars. Como fuente de datos deberá utilizarse la API Rest SWAPI o Star Wars API. El sitio web deberá estar disponible de forma pública.

URL SWAPI

<https://swapi.co/>

Navegación

El usuario debe ser capaz de ingresar al sitio y completar el siguiente flujo:

1. En la página principal, se deben listar todas las películas de la saga, con su nombre, año de lanzamiento, director y productor(es), y número de episodio. Cada película debe dar la opción de ver más información respecto a esta misma.
2. Al hacer click en una película, se debe mostrar una nueva página con toda la información que entrega la API relacionada a la película, filtrando la información de vehículos y especies.
3. La información que entrega la API de personas, naves y planetas, deberá mostrarse con el nombre de cada ítem, y no en la forma de url que entrega la llamada inicial de la API. La información sobre vehículos y especies no debe ser mostrada.
4. Al hacer click en cada uno de los elementos de la API (películas, personas, naves y planetas), deberá llevar a una nueva página que muestre toda la información de ese elemento en particular.

Búsqueda

En todo el sitio deberá mostrarse una barra de búsqueda, que al ingresar un texto, busque sobre los diferentes elementos de la API (películas, personas, naves y planetas).

Al hacer un click sobre el resultado de búsqueda, el sitio debe mostrar toda la información adicional disponible para el elemento seleccionado.

Otros requisitos

Arquitectura de la solución

El sitio web podrá estar construido sobre un framework MVC tal como Rails (Ruby), Django (Python), Sails (Nodejs) u otro, o como un sitio estático que consuma directamente la API.

En el primer caso, se recomienda hacer un deploy en Heroku, sistema de servidores Cloud que tiene una capa gratuita con capacidades suficientes para el desarrollo de esta tarea.

En el segundo caso, se puede utilizar un servicio como Github Pages o similar para publicar el sitio.

Uso de librerías helper

No se podrá utilizar ninguna librería helper de Swapi para acceder a la información. Esta restricción incluye las librerías oficiales publicadas en la sección Helper Libraries de documentación de Swapi como librerías no oficiales y no publicadas en el sitio.

Versionamiento del código

El sitio y todo su código fuente deberá estar versionado en un repositorio en Github.

Evaluación

Entregables

Cada alumno deberá entregar, mediante un formulario publicado en el sitio del curso, las siguientes *url's*:

- URL de inicio del sitio implementado
- URL del repositorio Github

Fecha de entrega

La tarea se deberá entregar el día 25 de marzo antes de las 15:30 hrs.

Requisitos mínimos

Las tareas que no cumplan las siguientes condiciones no serán corregidas y serán evaluados con la nota mínima:

- El sitio deberá ser público, accesible desde cualquier dispositivo conectado a internet.
- El código deberá estar versionado en su totalidad en un repositorio Git
- El sitio debe reflejar fielmente el código entregado en el repositorio
 - Se recomienda configurar un deploy automático en Heroku asociado al repositorio Github

Penalizaciones

Se descontará 1 punto de la nota de la tarea por cada hora de atraso en la entrega, contados a partir de la fecha estipulada en el punto anterior.

Cualquier intento de copia, plagio o acto deshonesto en el desarrollo de la tarea, será penalizado con nota 1,1 de acuerdo a la política de integridad académica del DCC.

Anexo 1: Recursos adicionales

A continuación, se presentan una serie de links que le podrán ser útiles para el desarrollo de la tarea.

Cursos sobre Ruby y Ruby on Rails

Rubymonk

Curso para aprender Ruby desde cero. Incluye ejemplos y una consola interactiva para ir ensayando y realizando ejemplos en el mismo navegador.

<https://rubymonk.com/>

Learn Ruby the Hard Way

Saltándote los ejercicios, este curso se puede completar en sólo 4 horas. El sitio también incluye cursos para otros lenguajes.

<https://learnrubythehardway.org/book/>

Codecademy

Curso que consiste en crear 8 aplicaciones en Rails, comenzando con lo más básico (arquitectura MVC, bases de datos, etc). Algunas partes de este curso son pagadas.

<https://www.codecademy.com/learn/learn-rails>

CodeSchool - Rails for Zombies

Sin duda, el curso más popular para aprender Rails. La versión gratuita del curso cubre todos los básicos y se completa en 2 a 3 horas según el sitio.

<https://www.codeschool.com/courses/rails-for-zombies-redux>

Creando un proyecto en Rails

Getting started with Rails

Guía oficial del *framework* para inicializar un proyecto utilizando Rails. Incluye todos los pasos para crear un proyecto, inicializar la aplicación, crear un layout y un modelo básico utilizando el ORM del *framework*.

Clean Architecture in Ruby

Artículo que muestra cómo crear un proyecto en Rails usando una arquitectura MVC.

<https://medium.com/@fbzga/clean-architecture-in-ruby-v2-d2293c78d4a6>

Resolviendo dudas

StackOverflow

Foro más popular para resolver dudas sobre código, *frameworks* y lenguajes de programación. Antes de preguntar, asegúrate que la pregunta no se ha realizado anteriormente.

<http://stackoverflow.com/>