Contenido

[2. Descripción Web Framework 3](#_Toc445763206)

[3. Fuentes de información 3](#_Toc445763207)

[3.1 Fuentes sobre Web Server Framework 3](#_Toc445763208)

[**3.1.1 Wikipedia** 3](#_Toc445763209)

[**3.1.2 Universidad de Sevilla /Javier J. Gutiérrez.** 4](#_Toc445763210)

[**3.1.3 http://kokiing.blogspot.com.es** 4](#_Toc445763211)

[3.2 Fuentes sobre Django 4](#_Toc445763212)

[**3.2.1 Wikipedia** 4](#_Toc445763213)

[**3.2.2 http://tutorial.djangogirls.org/es/django/** 5](#_Toc445763214)

[**3.2.3 http://computerhoy.com** 6](#_Toc445763215)

[**Django está de moda. Aunque sus orígenes se remontan a un diario online en 2003, por alguna razón ha sido en los últimos años cuando ha adquirido relevancia.** 6](#_Toc445763216)

[3.3 Fuentes sobre Node.js 6](#_Toc445763217)

[**3.3.1 Wikipedia** 6](#_Toc445763218)

[**3.3.2 http://www.netconsulting.es/blog/nodejs/** 7](#_Toc445763219)

[**3.3.3 http://www.genbetadev.com/frameworks/como-funciona-node-js** 7](#_Toc445763220)

[4. Fuentes de información (cursos no gratuitos) 8](#_Toc445763221)

[4.1 Cursos no gratuitos sobre otros web framework 8](#_Toc445763222)

[4.1.1 Curso no gratuito sobre Ruby on rails 8](#_Toc445763223)

[4.1.2 Curso no gratuito sobre Spring 8](#_Toc445763224)

[4.1.3 Curso no gratuito n sobre el tipo de tecnología en general 9](#_Toc445763225)

[4.2 Cursos no gratuitos Node.js 10](#_Toc445763226)

[4.2.1 Curso no gratuito 1 sobre Node.js 10](#_Toc445763227)

[4.2.2 Curso no gratuito 2 sobre Node.js 11](#_Toc445763228)

[4.2.3.Curso no gratuito 3 sobre Node.js 11](#_Toc445763229)

[4.3 Cursos no gratuitos Django 12](#_Toc445763230)

[4.3.1 Curso no gratuito 1 sobre Django 12](#_Toc445763231)

[Chapter Summary 13](#_Toc445763232)

[4.3.2 Curso no gratuito 2 sobre Django 13](#_Toc445763233)

[4.3.3.Curso no gratuito 3 sobre Django 14](#_Toc445763234)

[5. Fuentes de información (cursos gratuitos) 15](#_Toc445763235)

[5.1 Cursos gratuitos sobre Web server framework 15](#_Toc445763236)

[5.1.1 Curso gratuito 1 sobre Bootstrap 15](#_Toc445763237)

[5.1.2 Curso gratuito 2 sobre Bootstrap 16](#_Toc445763238)

[5.1.3 Curso gratuito 3 sobre Bootstrap 16](#_Toc445763239)

[5.2 Cursos gratuitos sobre la tecnología Django 17](#_Toc445763240)

[5.2.1 Curso gratuito 1 sobre la tecnología Django 17](#_Toc445763241)

[5.2.2 Curso gratuito 2 sobre la tecnología Django 17](#_Toc445763242)

[5.2.3 Curso gratuito 3 sobre la tecnología Django 18](#_Toc445763243)

[5.3 Cursos gratuitos sobre la tecnología Node.js 19](#_Toc445763244)

[5.3.1 Curso gratuito 1 sobre la tecnología Node.js 19](#_Toc445763245)

[[](https://www.youtube.com/watch?v=vnRGAsZdhx0) 20](#_Toc445763246)

[5.3.2 Curso gratuito 2 sobre la tecnología Node.js 20](#_Toc445763247)

[5.3.3 Curso gratuito 3 sobre la tecnología Node.js 20](#_Toc445763248)

[8. Conclusiones 22](#_Toc445763249)

# 2. Descripción Web Framework

Para saber que es un framework para aplicaciones web, vamos a empezar definiendo un framework:

-Framework: Estructura conceptual y tecnológica que sirve como base para la organización y desarrollo de software

Tomando esta definición, puede aplicarse un framework específico para proyectos de desarrollo web, consiguiendo un apoyo para el desarrollo de sitios web dinámicos, aplicaciones web y servicios web. Aliviando el exceso de carga asociado con actividades comunes, como puede ser la seguridad, la usabilidad, acceso a la base de datos, reutilización de código, etc.

En este documento se va a hablar de dos tecnologías específicas, Django y Node.js.

* Django: Framework de desarrollo web a alto nivel basado en Python. Unab ase para el desarrollo fácil, rápido y pragmático. Construido por desarrolladores experimentados.
* Node.js: entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript

# 3. Fuentes de información

En los sub-apartados de este apartado se deben indicar documentos de interés para aprender sobre el tipo de tecnología en general, y sobre cada una de las tecnologías elegidas.

## 3.1 Fuentes sobre Web Server Framework

### **3.1.1 Wikipedia**

Un framework para aplicaciones web es un framework diseñado para apoyar el desarrollo de [sitios web](https://es.wikipedia.org/wiki/Sitio_web) dinámicos, [aplicaciones web](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web) y [servicios web](https://es.wikipedia.org/wiki/Servicio_web). Este tipo de frameworks intenta aliviar el exceso de carga asociado con actividades comunes usadas en desarrollos web. Muchos framework proporcionan [bibliotecas](https://es.wikipedia.org/wiki/Biblioteca_%28inform%C3%A1tica%29) para acceder a bases de datos, estructuras para plantillas y gestión de sesiones, y con frecuencia facilitan la [reutilización de código](https://es.wikipedia.org/wiki/Reutilizaci%C3%B3n_de_c%C3%B3digo).

Para proporcionar una página web dinámica que refleja las entradas del usuario , se introdujo la norma ( CGI ) Common Gateway Interface para conectar aplicaciones externas con los servidores web. CGI podría afectar negativamente a la carga del servidor , sin embargo ,cada solicitud inicia un proceso separado .

Muchos marcos siguen el patrón de arquitectura MVC para separar el modelo de datos con reglas de negocio de la interfaz de usuario

### **3.1.2 Universidad de Sevilla /Javier J. Gutiérrez.**

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Un framework Web, por tanto, podemos definirlo como un conjunto de componentes (por ejemplo clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web. Para comprender como trabajan los frameworks Web existentes es imprescindible conocer el patrón MVC.

El patrón Modelo-Vista-Controlador es una guía para el diseño de arquitecturas de aplicaciones que ofrezcan una fuerte interactividad con usuarios. Este patrón organiza la aplicación en tres modelos separados, el primero es un modelo que representa los datos de la aplicación y sus reglas de negocio, el segundo es un conjunto de vistas que representa los formularios de entrada y salida de información, el tercero es un conjunto de controladores que procesa las peticiones de los usuarios y controla el flujo de ejecución del sistema.

### **3.1.3** [**http://kokiing.blogspot.com.es**](http://kokiing.blogspot.com.es)

El propósito de un Framework es ayudar y facilitar el proceso de desarrollo de aplicaciones. Debe permitir desarrollar la aplicación rápido y fácilmente y debe resultar en una aplicación superior finalizada.

Frameworks, en resumen, proveen una poderosa caja de herramientas. Las herramientas en esta caja ayudan en muchas áreas diferentes del desarrollo de aplicaciones. Proveen patrones de diseño esenciales y estructura al proyecto de desarrollo de la aplicación, y también provee el esqueleto y contenedor para los componentes que se crean para operar la aplicación.

Los Frameworks son importantes en todas las fases de desarrollo, desde el diseño hasta el desarrollo y quizá más en el mantenimiento continúo.

## 3.2 Fuentes sobre Django

### **3.2.1 Wikipedia**

Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo–vista–controlador. Fue desarrollado en origen para gestionar varias páginas orientadas a noticias de la World Company de Lawrence, Kansas, y fue liberada al público bajo una licencia BSD en julio de 2005; el framework fue nombrado en alusión al guitarrista de jazz gitano Django Reinhardt.

En junio de 2008 fue anunciado que la recién formada Django Software Foundation se haría cargo de Django en el futuro.

La meta fundamental de Django es facilitar la creación de sitios web complejos. Django pone énfasis en el re-uso, la conectividad y extensibilidad de componentes, el desarrollo rápido y el principio No te repitas (DRY, del inglés *Don't Repeat Yourself*). Python es usado en todas las partes del framework, incluso en configuraciones, archivos, y en los modelos de datos.

Los orígenes de Django en la administración de páginas de noticias son evidentes en su diseño, ya que proporciona una serie de características que facilitan el desarrollo rápido de páginas orientadas a contenidos. Por ejemplo, en lugar de requerir que los desarrolladores escriban controladores y vistas para las áreas de administración de la página, Django proporciona una aplicación incorporada para administrar los contenidos, que puede incluirse como parte de cualquier página hecha con Django y que puede administrar varias páginas hechas con Django a partir de una misma instalación; la aplicación administrativa permite la creación, actualización y eliminación de objetos de contenido, llevando un registro de todas las acciones realizadas sobre cada uno, y proporciona una interfaz para administrar los usuarios y los grupos de usuarios (incluyendo una asignación detallada de permisos).

### **3.2.2** [**http://tutorial.djangogirls.org/es/django/**](http://tutorial.djangogirls.org/es/django/)

Django es un framework para aplicaciones web gratuito y open source, escrito en Python. Es un WEB framework - un conjunto de componentes que te ayudan a desarrollar sitios web más fácil y rápidamente.

Verás, cuando estás construyendo un sitio web, frecuentemente necesitas un conjunto de componentes similares: una manera de manejar la autenticación de usuarios (registrarse, iniciar sesión, cerrar sesión), un panel de administración para tu sitio web, formularios, una forma de subir archivos, etc.

Por suerte para ti, hace tiempo varias personas notaron que los desarrolladores web enfrentan problemas similares cuando construyen un sitio nuevo, por eso juntaron cabezas y crearon frameworks (Django es uno de ellos) que te ofrecen componentes listos para usarse.

Los frameworks existen para ahorrarte tener que reinventar la rueda y ayudarte a aliviar la carga cuando construyes un sitio.

Cuando llega una petición a un servidor web es pasado a Django que intenta averiguar lo que realmente es solicitado. Toma primero una dirección de página web y trata de averiguar qué hacer. Esta parte es realizada por urlresolver de Django (tenga en cuenta que la dirección de un sitio web es llamada URL - Uniform Resource Locator - así que el nombre *urlresolver* tiene sentido). Este no es muy inteligente - toma una lista de patrones y trata de igualar la URL. Django comprueba los patrones de arriba hacia abajo y si algo se coincide entonces Django le pasa la solicitud a la función asociada (que se llama *vista*).

### **3.2.3** [**http://computerhoy.com**](http://computerhoy.com)

### **Django está de moda. Aunque sus orígenes se remontan a un diario online en 2003, por alguna razón ha sido en los últimos años cuando ha adquirido relevancia.**

Si has intentando conocer un poco más acerca de Django, te habrás topado con la densa definición de la entradilla. En pocas palabras, resume muchas de las propiedades de este proyecto, así que veamos de qué se trata.

Proporciona una serie de herramientas para facilitar la creación de páginas, siguiendo los principios DRY (Don’t Repeat Yourself; No Te Repitas) para evitar duplicidad en las líneas de código e invertir el menor esfuerzo posible. Por ejemplo, levantar un panel de administración básico sólo requiere un par de líneas de Python.

También se adscribe al diseño MVC (Modelo-Vista-Controlador), por lo que las diferentes partes del sitio están claramente separadas. Por ejemplo, el código de acceso a los datos es completamente independiente al que gobierna el aspecto externo de la página.

Al tratarse de Python, Django permite que el desarrollador escriba código ágilmente. El resultado son menos líneas de código y, en consecuencia, menos probabilidades de que haya bugs. De ahí que digan que “fomenta el desarrollo rápido y el diseño limpio y pragmático”.

Django no pretende sustituir a WordPress como CMS. Sólo está ahí para ayudarte a levantar sitios web desde cero, sea del tipo que sea. En vista de la versatilidad forzosa a la que está siendo sometido WordPress últimamente, quizá quieras optar por Django en busca de la flexibilidad necesaria para cualquier tipo de proyecto.

Aprenderás Python de una forma práctica y limpia, pues la implementación de Django obliga a respetar muchos principios de la buena programación.

## 3.3 Fuentes sobre Node.js

### **3.3.1 Wikipedia**

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web.[2](https://es.wikipedia.org/wiki/Node.js#cite_note-2) Fue creado por Ryan Dahl en 2009 y su evolución está apadrinada por la empresa Joyent, que además tiene contratado a Dahl en plantilla.

Node.js es similar en su propósito a Twisted o Tornado de Python, Perl Object Environment de Perl, libevent o libev de [C](https://es.wikipedia.org/wiki/C), EventMachine de Ruby, vibe.d de Dy JEE de Java existe Apache MINA, Netty, Akka, Vert.x, Grizzly o Xsocket. Al contrario que la mayoría del código JavaScript, no se ejecuta en un navegador, sino en el servidor. Node.js implementa algunas especificaciones de CommonJS.[5](https://es.wikipedia.org/wiki/Node.js#cite_note-5) Node.js incluye un entorno REPL para depuración interactiva.

Node.js funciona con un modelo de evaluación de un único hilo de ejecución, usando entradas y salidas asíncronas las cuales pueden ejecutarse concurrentemente en un número de hasta cientos de miles sin incurrir en costos asociados al cambio de contexto.[6](https://es.wikipedia.org/wiki/Node.js#cite_note-6) Este diseño de compartir un único hilo de ejecución entre todas las solicitudes atiende a necesidades de aplicaciones altamente concurrentes, en el que toda operación que realice entradas y salidas debe tener una función callback. Un inconveniente de este enfoque de único hilo de ejecución es que Node.js requiere de módulos adicionales como *cluster*para escalar la aplicación con el número de núcleos de procesamiento de la máquina en la que se ejecuta.

### **3.3.2** [**http://www.netconsulting.es/blog/nodejs/**](http://www.netconsulting.es/blog/nodejs/)

Node.js es una librería y entorno de ejecución de E/S dirigida por eventos y por lo tanto asíncrona que se ejecuta sobre el intérprete de JavaScript creado por Google V8. Lo cierto es que está muy de moda aunque no es algo nuevo puesto que existen librerías como Twisted que hacen exactamente lo mismo pero si es cierto que es la primera basada en JavaScript y que tiene un gran rendimiento.

Node.js es un entorno Javascript del lado del servidor, basado en eventos. Node ejecuta javascript utilizando el motor V8, desarrollado por Google para uso de su navegador Chrome. Aprovechando el motor V8 permite a Node proporciona un entorno de ejecución del lado del servidor que compila y ejecuta javascript a velocidades increíbles. El aumento de velocidad es importante debido a que V8 compila Javascript en código de máquina nativo, en lugar de interpretarlo o ejecutarlo como bytecode. Node es de código abierto, y se ejecuta en Mac OS X, Windows y Linux.

Aunque Javascript tradicionalmente ha sido relegado a realizar tareas menores en el navegador, es actualmente un lenguaje de programación totalmente, tan capaz como cualquier otro lenguaje tradicional como C++, Ruby o Java. Además Javascript tiene la ventaja de poseer un excelente modelo de eventos, ideal para la programación asíncrona. Javascript también es un lenguaje omnipresente, conocido por millones de desarrolladores. Esto reduce la curva de aprendizaje de Node,js, ya que la mayoría de los desarrolladores no tendrán que aprender un nuevo lenguaje para empezar a construir aplicaciones usando Node.js.

### **3.3.3** [**http://www.genbetadev.com/frameworks/como-funciona-node-js**](http://www.genbetadev.com/frameworks/como-funciona-node-js)

Todos hemos oído hablar de Node.js como tecnología que permite trabajar con Javascript del lado del servidor. Poco a poco Node se está ganando un hueco en la comunidad. Sin embargo hay muchas dudas sobre su funcionamiento. Vamos a intentar en este artículo acercarnos un poco más a esta tecnología tan interesante.

Node.js está basado en el motor V8 de Javascript de Google. Este motor está diseñado para correr en un navegador y ejecutar el código de Javascript de una forma extremadamente rápida.

La tecnología que está detrás de Node.js permite ejecutar este motor en el lado del servidor abriendo un nuevo abanico de posibilidades en cuanto al mundo de desarrollo se refiere.

Para ello el entorno de Node.js ha desarrollado un conjunto amplio de librerías que no tienen nada que envidiar a otras plataformas. Por otro lado se han eliminado algunas funcionalidades que en el entorno de servidor no tenían sentido como por ejemplo el uso de Document Object Model.

Para poder trabajar de una forma óptima Node.js delega todo el trabajo en un pool de threads. Este pool de threads está construido con la librería libuv. Esta librería dispone de su propio entorno multithread asíncrono. Node.js envía el trabajo que hay que realizar al pool.

# 4. Fuentes de información (cursos no gratuitos)

## 4.1 Cursos no gratuitos sobre otros web framework

### 4.1.1 Curso no gratuito sobre Ruby on rails

Ruby on rails es un framework que te ayuda a construir aplicaciones para la web.



* Link: <http://code.tutsplus.com/courses/get-started-with-ruby-on-rails>
* Duración: 2.4horas
* Precio: 9$
* Lecciones: 14

En este curso orientado a principiantes en ruby, aprenderás construyendo una aplicación cubriendo lo básico de Ruby: modelos, controladoras y vistas.

También aprenderás a realizar validaciones, migraciones, envíos de mail, integración con peticiones AJAX, etc.

### 4.1.2 Curso no gratuito sobre Spring



* Link: <http://www.edureka.co/spring-framework>
* Precio: 296$
* Requiere conocimientos básicos de Java y bases de datos

Spring es el mas famoso framework de java, spring combina el modelo MVC y el mapeo de la base de datos en un solo paquete.

Spring proporciona Programación orientada a aspectos, soporte para el testeo, inyección de dependencias, etc.

Este curso que cubre todos los conceptos del popular framework de Java, integramiento con hibernate, maven, uso de MVC, Apache, etc.

Objetivos del curso:

1. Spring Framework Architecture

2. Spring's Dependency Injection and Autowiring feature

3. Build application using Maven

4. Aspect Oriented Programming (AOP) and AspectJ

5. Data access mechanisms provided by Spring

6. Integrating Hibernate with Spring

7. Spring MVC

8. Create a web application using Spring MVC

9. Integrating Apache tiles with Spring

10. Integrating Struts2 and JSF with Spring

11. Spring Web Flow

12. Keeping your application secure

13. Using log4j for logging

14. Writing test cases with JUnit

15. Spring Integration Framework

### 4.1.3 Curso no gratuito n sobre el tipo de tecnología en general



Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja

Symfony está desarrollado completamente con PHP, es fácil de instalar y de configurar en la mayoría de las plataformas.

* Link: <https://www.udemy.com/symfony-development/>
* Clases: 20
* Videos: 1 hora 30 minutos
* Requisitos: conocimientos en HTML, CSS, JS, PHP y MySQL

Temario que se imparte:

* What is Symfony and what are the benefits of using it in your webapps
* Installation and new project creation.
* Controllers. How to send back HTML or JSON to the user.
* Routing URLs to specific actions.
* View and the Twig templating language.
* Intro to Doctrine ORM.
* Creating a simple CRUD webapp.
* Including external assets in your projects.

## 4.2 Cursos no gratuitos Node.js

### 4.2.1 Curso no gratuito 1 sobre Node.js

* Link: <http://www.fictizia.com/formacion/curso_nodejs?gclid=Cj0KEQjwwpm3BRDuh5awn4qJpLwBEiQAATTAQcwRZd26_PrtPB801w_vP4Sh6ZE3wV9p8lJhnPr4ITQaAtSf8P8HAQ>
* Duración: 24h
* Precio: 490€
* Orientado a principiantes

El programa del Curso está diseñado para que los alumnos descubran la bases y fundamentos del desarrollo con JavaScript en el lado del servidor y adquieran las capacidades necesarias para automatizar tareas del día a día como la minificación y el testing.

Los alumnos aprenderán también todos los conceptos fundamentales del desarrollo con APIs modernas para compartir datos con otros de una manera eficiente (AJAX server-side), el uso de la terminal, la asincronía y peculiaridades propias de Node.

Con este curso aprenderás a:

* Desarrollar aplicaciones web en tiempo real con Firebase (WebSocket).
* Comprender en profundidad el protocolo HTTP
* Enriquecer fácilmente cualquier proyecto web con la inclusión de APIs.
* Conocer cómo trabajar con un control de versiones como Git.
* Tener la capacidad de contribuir a proyectos de código abierto usando GitHub.
* Integrarte en proyectos Full-Stack donde se requiera el uso de frameworks de testing.
* Aprender cómo crear proyectos interactivos que requieran de un API Rest.

### 4.2.2 Curso no gratuito 2 sobre Node.js

* Link: <http://www.educacionit.com/curso-de-nodejs-y-mongodb?gclid=Cj0KEQjwwpm3BRDuh5awn4qJpLwBEiQAATTAQTP-vTBrQXanrjdKELkgIr-ViVyoGQrjKSmGhVpX61YaAh5U8P8HAQ>
* Requisitos: Experiencia en lenguajes servidor (PHP, java, python, etc)
* Precio: 235€
* Duración: 18h

Los estudiantes aprenderán a diseñar y desarrollar soluciones Web, como así también la puesta en producción de sistemas basados en Node.js sobre servidores Linux (Debian)

La finalidad de este curso es desarrollar prácticas de casos reales en un ambiente de trabajo profesional. Es por eso que también se utilizara el framework Express.js que es un framework extensible de manejo de servidores HTTP, el cual provee plugins de alto rendimiento conocidos como middleware sumamente utilizado en grandes proyectos con Node.js

### 4.2.3.Curso no gratuito 3 sobre Node.js

* Link del curso: <https://www.udemy.com/the-complete-node-js-developer-course/>
* 142 clases
* 27 horas de video
* No es necesaria ninguna experiencia en programación
* Precio: 27€ con promoción.

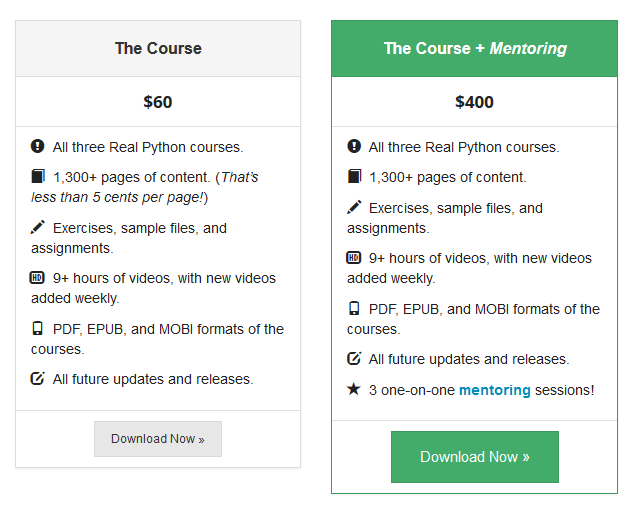
Un curso muy completo en el que construirás 5 aplicaciones reales, todo desde el principio. Desde principiante hasta desarrollador de back-end.

Cada video viene con un ejercicio de programación, para que puedas ver tu curva de aprendizaje.

Una vez tengas los conocimientos fundamentales añadirás módulos como Express, Sequelize o SocketIO.

## 4.3 Cursos no gratuitos Django

### 4.3.1 Curso no gratuito 1 sobre Django

* Link del curso: <https://realpython.com/courses/#course-3-advanced-web-development-with-django>
* Este curso se divide en 3 sub-cursos
  + Introducción a Python
  + Desarrollo web con Python
  + Desarrollo web con django avanzado
* Nivel del curso: Orientado a programadores expertos en Python, o que hayan hecho los dos sub-cursos anteriores.
* Precio: 

En este curso se abarca el aprendizaje de Python y de django en 3 distintos sub-cursos, ya que como dijimos, este último está basado en ese famoso lenguaje de programación. Si por cualquier motivo quieres saltarte los sub-cursos preliminares de introducción, no pasa nada, puedes ir directamente al curso tercero del link, en el que se trata directamente a django, aunque es recomendado hacerlos todos.

Índice del sub-curso 3, dedicado a django:

### Chapter Summary

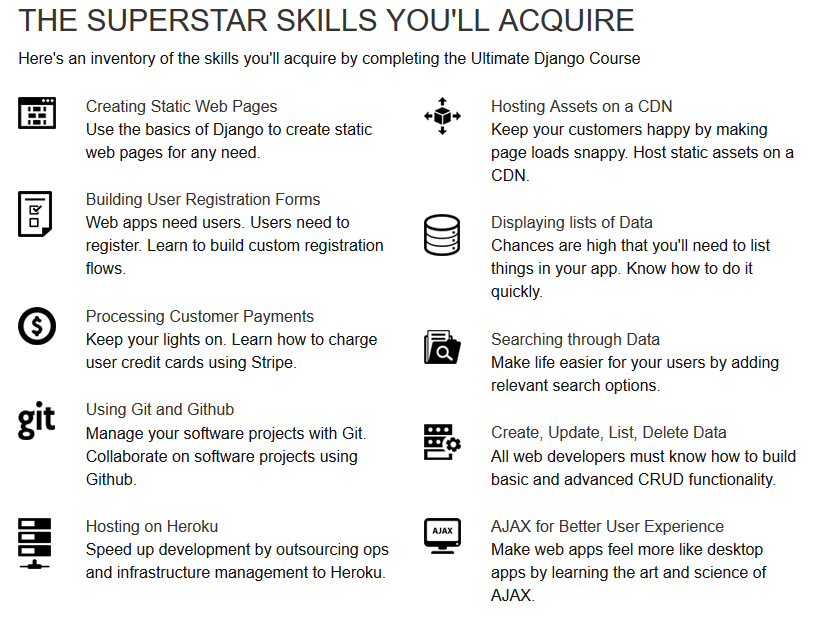
1. Preface
2. Introduction
3. Software Craftsmanship
4. Test Driven Development
5. Git Branching at a Glance
6. Upgrade, Upgrade, and Upgrade some more
7. Graceful Degradation and Database Transactions with Django 1.8
8. Building a Membership Site
9. Bootstrap 3 and Best Effort Design
10. Building the Members Page
11. REST
12. Django Migrations
13. AngularJS Primer
14. Djangular: Integrating Django and Angular
15. Angular Forms
16. MongoDB Time!
17. One Admin to Rule Them All
18. Testing, Testing, and More Testing
19. Deploy
20. Conclusion
21. Appendix A – Solutions to Exercises

### 4.3.2 Curso no gratuito 2 sobre Django

* Link del curso: <https://ultimatedjango.com/>
* Tiempo necesario: Si estudias 5 horas semanales, se completa en 6-8 semanas
* Precio: 29$ la starter edition, 79$ la profesional.

La versión estándar incluye 17 capítulos, una sección de Q&A en el foro, el eBook de ultimade django y una referencia de código en git.

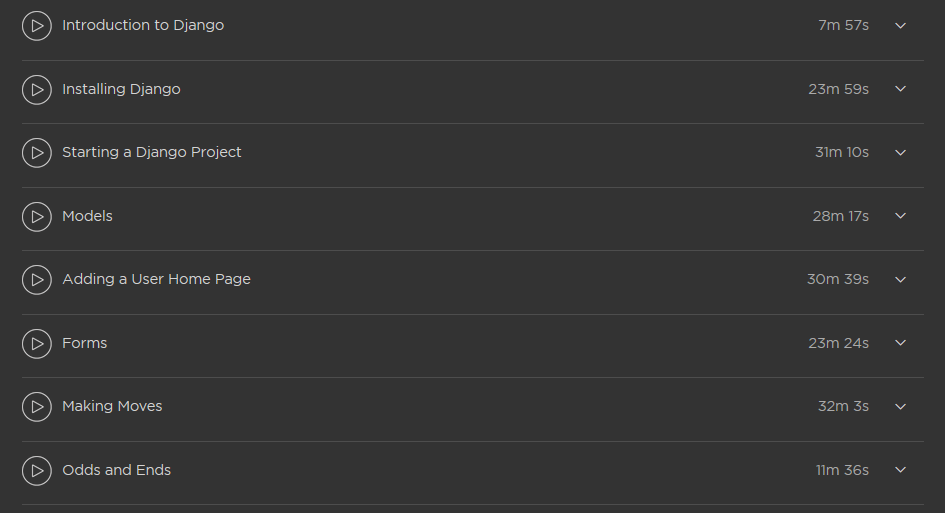
La versión profesional incluye lo citad oanteriormente y además una guía de Angular/Django, React/Django, Backbone/Django y de como desarrollar aplicaciones RESTful en django.



### 4.3.3.Curso no gratuito 3 sobre Django

* Link: <https://www.pluralsight.com/courses/django-fundamentals>
* Curso de pluralsight, puedes acceder a sus cursos con una suscripción anual.
* Precio de la suscripción:
  + 299$ (24$.92/mes) por la estándar
  + 499$ (41.58/mes) la suscripción plus
* Duración del curso: aproximadamente 3h

Esta es la tabla de contenidos del curso:



# 5. Fuentes de información (cursos gratuitos)

## 5.1 Cursos gratuitos sobre Web server framework

### 5.1.1 Curso gratuito 1 sobre Bootstrap

Link: <https://aula.tareasplus.com/Yan-Arlex-Vallejo-Rosas/Disena-proyectos-web-Bootstrap/Bootstrap-3---Clase-1---Introduccion>

Este segundo curso trata sobre bootstrap 3, se encuentra en la página web *Tareas Plus Para saber más* para acceder al curso te tienes que registrar o entrar con una cuenta de google o Facebook, el registro es totalmente gratuito. Las lecciones de dividen en 18 videos que tratan los siguientes temas:

1. Introducción
2. Nuestro primer contenido
3. Sistema de grillas
4. Grillas – parte 1
5. Grillas – parte 2
6. Grillas – parte 3 (Final)
7. Tipografía, tablas
8. Formularios – parte 1
9. Formularios – parte 2
10. Botones
11. Imágenes
12. Clases de ayuda
13. Glyphicons
14. Dropdowns
15. Grupos de botones
16. Botones desplegables
17. Grupos de input
18. Navs

La duración de cada video es de unos 20 minutos aproximadamente. Este curso se puede interactuar con otros estudiantes y exponer sus dudas.

### 5.1.2 Curso gratuito 2 sobre Bootstrap

Link: <http://www.aulaclic.es/bootstrap-en-videos/001-bootstrap_preparandose.htm>

Este curso trata sobre Bootstrap, que permite crear sitios web y aplicaciones web totalmente responsive sin necesidad de complicaciones en el desarrollo. En este curso intensivo se da un enfoque a twitter bootstrap, en él se puede aprender todas las funcionalidades sobre los elementos que queramos y el diseño de estos (cabeceras, márgenes, responsive…).Se compone de 12 lecciones todas ellas en videos de una media de 10 min de duración. Se encuentra en la página aula clic que trata de una web en español que contiene cursos de informática.

Este curso Intensivo de Twitter Bootstrap se divide en las siguientes lecciones:

1. Preparándose.
2. Convertir enlaces en botones
3. Sistema de diseño
4. Scaffolding
5. Uso de la clase Thumbnails
6. Crear grids
7. El navbar
8. Uso de JavaScript para hacer el diseño Responsive
9. Convertir proyecto de bootstrap 2 a bootstrap 3
10. Guía de novedades en bootstrap 3
11. Crear pestañas
12. Crear un slideshow.

### 5.1.3 Curso gratuito 3 sobre Bootstrap

Link: <https://librosweb.es/libro/bootstrap_3/>

Este curso se encuentra en la página web librosweb.es se trata del manual oficial de bootstrap se compone de 7 capítulos en donde se tratan los siguientes temas:

* Capítulo 1 primeros pasos: en este capítulo se explica los contenidos de bootstrap, la compatibilidad que tiene con los navegadores además de la licencia y accesibilidad entre otros temas.
* Capítulo 2 diseñando con rejilla: se trata de los tipos de rejilla y como trabajar con columnas.
* Capítulo 3 tipografía: se aprende sobre listas, clases css, énfasis, etc.
* Capítulo 4 elementos CSS: tablas, imágenes y utilidades.
* Capítulo 5 formularios: se trata sobre los distintos tipos de formularios además del redimensionado de estos y los botones.
* Capítulo 6 componentes: en este capítulo se aprende sobre los componentes de bootstrap como pueden ser iconos, paneles, objetos multimedia, elementos de navegación, etc.

Al ser un manual este curso no podemos especificar la duración ya que además se divide en varios temas cada capítulo, estudiarlo conllevaría un tiempo extenso pero interesante a la vez para todo aquel que le interese el diseño web.

## 5.2 Cursos gratuitos sobre la tecnología Django

### 5.2.1 Curso gratuito 1 sobre la tecnología Django

Link: <https://codigofacilito.com/videos/curso_django_como_instalar_django>

Se trata de un curso que comprende 29 lecciones sobre Django, framework para aplicaciones web gratuitos y open source escrito en Python. A través de estos videos tutoriales se enseña a como instalar Django o como realizar nuestro primer proyecto. También nos dan la opción de registrarnos gratuitamente y generar todas las dudas que tengamos para que nos la resuelvan. Este curso se encuentra en la página web codigofacilito.com, calculamos que la duración media por cada lección esta entre 5 -6 min.

Algunos de los temas que destacan es este curso son:

1. API Shell
2. Configurar URL’s para imágenes
3. Cache en Django
4. Las relaciones
5. Como instalar PIP

Entre otro más que nos ayudan a crear nuestro primer modelo y a trabajar con bases de datos.

### 5.2.2 Curso gratuito 2 sobre la tecnología Django

Link: <https://edutin.com/clases/online/41005185>

El siguiente curso de video tutoriales totalmente gratuitos fue creado por Jesús Conde Núñez, creador del Portal Educativo Videotutoriales.com. Se accede a través de google con el Gmail. La duración total de este curso es de 30 minutos repartidos en 21 clases que contiene la siguiente información:

1. ¿Qué es y cómo instalarlo?
2. Análisis de un proyecto Django
3. Bases de datos
4. Creación y uso de modelos
5. Uso de la API Database
6. URLconf y Views
7. Añadir datos a Views desde Base Datos
8. Views, Exceptions y Shortcuts
9. Uso de Templates
10. Crear segunda Template
11. Imports y herencias en templates
12. El sitio de administración de Django
13. Personalizar panel de administración
14. Uso de formularios
15. Persistir datos de formularios
16. Uso de modelForm
17. Crear página para editar preguntas
18. Personalizar estilos formulario
19. Crear sistema de login
20. Crear página de inicio
21. Restringir acceso a paginas

### 5.2.3 Curso gratuito 3 sobre la tecnología Django

Link: <http://codehero.co/django-desde-cero-modelos-y-base-de-datos/>

Se trata de un curso rápido y sencillo con una duración de 25 minutos para principiantes con el uso de Django. Abarca menos temas que los anteriores pero el aprendizaje es más dinámico ya que se trata de un manual en el que nos proporciona ejemplos además de código ya picado para facilitarnos el trabajo.

1. Instalación y herramientas
2. [Modelos y Base de Datos](http://codehero.co/django-desde-cero-modelos-y-base-de-datos)
3. [Vistas Dinámicas](http://codehero.co/django-desde-cero-vistas-dinamicas)
4. [Sitio de Administración](http://codehero.co/django-desde-cero-sitio-de-administracion)
5. [Formularios (Forms)](http://codehero.co/django-desde-cero-formularios-forms)
6. [Urls Avanzadas](http://codehero.co/django-desde-cero-urls-avanzadas)
7. [Validaciones de formulario](http://codehero.co/django-desde-cero-validaciones-de-formulario)
8. [Relaciones o Asociaciones en Base de Datos (Parte I)](http://codehero.co/django-desde-cero-relaciones-o-asociaciones-en-base-de-datos-parte-i)
9. [Relaciones o Asociaciones en Base de Datos (Parte II)](http://codehero.co/relaciones-o-asociaciones-en-base-de-datos-parte-ii)

## 5.3 Cursos gratuitos sobre la tecnología Node.js

### 5.3.1 Curso gratuito 1 sobre la tecnología Node.js

Link: <https://www.maestrodelacomputacion.net/curso-de-nodejs-gratuito-y-en-espanol-videotutoriales/>

En este curso que se encuentra en la página web Mestro de la Computación Informática al alcance de todos, trata sobre la tecnología nodejs. Se divide en 29 videos tutoriales subidos a YouTube. Antes de comenzar este curso gratuito la persona debería de tener un poco de idea del Javascript, de esta manera aprender le será más fácil teniendo esa base. Los temas que se abordan son los siguientes:

1. Introducción a node.js
2. Node interactivo con REPL
3. Atajos de teclado y comando de REPL
4. El Core de Node
5. Servers, Strems y Sockets.
6. Crear Aplicaciones Node.js con WebMatrix
7. Routing
8. Servir archivos estáticos
9. Almacenar contenidos en la cache
10. Optimizar rendimiento con el uso de Streaming
11. Evitar Hacking Exploits en archivos estáticos
12. Objeto HTTP. Hacer
13. Gestionar subidas de archivos
14. Cliente HTTP y Sniffer
15. Instalar y usra el framework Express
16. Uso de Jade, motor de plantillas para Node.js
17. Routing y Renderizado de Views para Node.js
18. Persistencia de datos
19. Instalación de MongoDB y Mongoose
20. Creacion de aplicación CRUD.Vistas Index y Crear
21. Creación de Aplicación CRUD. Vistas Edit y Delete
22. Ajax bidireccional con Socket.IO
23. Comunicaciones Bidireccionales cliente – servidor
24. Creación de un Servidor de Chat Socket.IO
25. Servidor de Chat Socket.IO (II)
26. Cliente de Twitter en Streaming.
27. Creación aplicación Twitter completa.
28. Uso de MySQL con node-mysql.
29. Publicar nuestros proyectos

La duración de cada video tutorial tiene de media unos 30 minutos aproximadamente está en un español castellano apto para todo usuario interesado en el aprendizaje de Node.js

### 

### 5.3.2 Curso gratuito 2 sobre la tecnología Node.js

Link: <http://www.tutorialspoint.com/nodejs/nodejs_tutorial.pdf>

Se trata de un manual totalmente gratuito en formato pdf apto para su descarga e impresión. Consta de 169 páginas redactadas en ingles con distintos apartados en los que se abordan temas como por el ejemplo que es node.js, para que sirve, como usarlo, instalación en Windows, los conceptos de node.js, etc. Es un manual sencillo e interactivo para alguien que se quiere iniciar en este mundo de node.js, la única pega que tiene es el idioma ya que se encuentra en inglés.

El tutorial lo podemos encontrar online también en el siguiente enlace:

<http://www.tutorialspoint.com//nodejs/index.htm>

### 5.3.3 Curso gratuito 3 sobre la tecnología Node.js

Link: <https://www.youtube.com/watch?v=61ybknbAhuA&index=4&list=PLJEglAUdzLXIvSSK6oZwZ0VmZKSFYFgx0>

El último curso de esta tecnología trata de 19 video tutoriales que van paso a paso comenzando por cómo es la instalación de node.js, son unos videos tutoriales creados por Josematube que se trata de un canal lleno de tutoriales de arendizaje.Los videos son cortos de una duración media de 5 minutos además de aclarativos ya que va explicando los conceptos al mismo tiempo que enseña.los temas que aborda son :

1. Instalacion de node
2. Eligiendo un editor de texto
3. Creando nuestra orimera “clase” en javascript
4. Herencia de la “clase” básica
5. Operaciones básicas con arrays en javascript
6. Funciones shift y unshift en javascript
7. Verificar si es array en javascript
8. Búsquedas sencillas en arrays
9. Filtrar un array
10. Recorrer cada elemento de un array
11. Obtener sub array de un array
12. Ejemplo objeto JSON
13. Objeto JSON con array
14. Ejemplo de Singleton
15. Modulos y requires de nodejs
16. Eventos en nodejs
17. Depuración en nodejs
18. Npm o node package manager
19. Agrupar modulos en nodejs

Con este sencillo curso se aprende a llegar al usuario y al interés de este por node.js además de que aprendan y se inicie en este web framework.

[](https://www.youtube.com/watch?v=d2BPdyuUcyE)

# 6. Ayudas para estudiar las tecnologías

# 7. Recursos para implementar las tecnologías

## 7.1 Recursos para implementar la tecnología A

### 7.1.1 Recursos no gratuitos para implementar la tecnología A

En la parte de recursos no gratuitos, al igual que con Django, tenemos la contratación del hosting para albergar estos servicios, así como si lo ejecutamos en una máquina local, el coste que supone la luz, internet, IP estática, la máquina en si, etc.

### 7.1.1 Recursos gratuitos para implementar la tecnología A

Dentro de los recursos gratuitos para la implementación de Nodejs, existe la documentación de su página oficial: <https://nodejs.org/en/docs/>

En este enlace podemos encontrar documentación sobre sus APIs, preguntas frecuentes sobre Nodejs y guías de instalación y demás.

Usando Google, podemos encontrar una inmensa cantidad de información. Por ejemplo, podemos encontrar una buena guía de instalación en desarrolloweb.com: <http://www.desarrolloweb.com/articulos/instalar-node-js.html>

## 7.2 Recursos para implementar la tecnología B

### 7.2.1 Recursos no gratuitos para implementar la tecnología B

Para la implementación de Django es necesario disponer de una máquina servidor, ya sea un servidor dedicado, un VPS o un servidor local. Esto puede variar desde 5€/mes hasta lo que se quiera o necesite pagar. Esto depende del nivel de tráfico que tenga nuestra web.

### 7.2.1 Recursos gratuitos para implementar la tecnología B

Además de los recursos no gratuitos ya nombrados arriba, Django necesita una aplicación servidor, como puede ser Apache.

Django también necesita Python para poder funcionar, además de servidores de bases de datos como puede ser MySQL o PostgreSQL.

Django dispone de documentación oficial en su página para poder realizar su instalación: <https://docs.djangoproject.com/en/1.9/intro/install/>

Además de un tutorial de instalación, esta documentación contiene 9 tutoriales para avanzar con esta tecnología:

* Request and responses: <https://docs.djangoproject.com/en/1.9/intro/tutorial01/>
* Models and the admin site: <https://docs.djangoproject.com/en/1.9/intro/tutorial02/>
* Views and templates: <https://docs.djangoproject.com/en/1.9/intro/tutorial03/>
* Forms and generic views: <https://docs.djangoproject.com/en/1.9/intro/tutorial04/>
* Testing: <https://docs.djangoproject.com/en/1.9/intro/tutorial05/>
* Static files: <https://docs.djangoproject.com/en/1.9/intro/tutorial06/>
* Customizing the admin site: <https://docs.djangoproject.com/en/1.9/intro/tutorial07/>

Además, tiene dos tutoriales algo más avanzados:

* How to write reusable apps: <https://docs.djangoproject.com/en/1.9/intro/reusable-apps/>
* Writing your first patch for Django: <https://docs.djangoproject.com/en/1.9/intro/contributing/>

# 

# 8. Conclusiones

Para finalizar el trabajo llegamos a  una conclusión común que se trata del aprendizaje que nos ha aportado sobre web server framework y las tecnologías que comparamos que son Django y Node.js. Anteriormente ya trabajamos en otras asignaturas con bootstrap por ejemplo por ello lo añadimos aquí en este trabajo ya que no gustó a todos la sencillez que tiene y el resultado final tan limpio que nos proporciona. Sobre estos dos frameworks en particular podemos decir que nos han ayudado a aprender sobre otro tipo de tecnología framework y tener un repertorio más amplio en un futuro ya que nos serán de gran ayuda.