



GRADO EN INGENIERÍA INFORMÁTICA

TRABAJO FINAL DE GRADO

E-Force Scraper

Autor:
Dragos Verdes

Supervisor:
Jose Antonio Cano Gómez

Tutor académico:
Vicente Cholvi Juan

Fecha de lectura: 09 de Septiembre de 2016

Curso académico: 2015/2016

Resumen:

En este documento se describe el proceso de desarrollo de un portal web encargado de realizar funciones de scraping. También se describe como se ha desarrollado la integración de otros servicios para dicha aplicación.

El web scraper se ha realizado durante la estancia en prácticas en la empresa EForce Spain, una empresa especializada en el desarrollo de páginas web dinámicas utilizando el framework ASP .NET .

Palabras claves:

ASP .NET, C#, MVC (Modelo-Vista-Controlador), Web Scraper, Jenkins, Slack, Integración

Key words:

ASP .NET, C#, MVC (Model-View-Controller), Web Scraper, Jenkins, Slack, Integration

Índice general:

| | | |
|--------|---|----|
| 1. | Introducción | 6 |
| 1.1. | Introducción al proyecto | 6 |
| 1.2. | Motivación | 6 |
| 1.3. | Objetivos | 7 |
| 2. | Descripción del proyecto | 9 |
| 2.1. | La empresa | 9 |
| 2.2. | Alcance | 9 |
| 2.3. | Tecnologías utilizadas | 10 |
| 3. | Planificación | 14 |
| 3.1. | Definición de tareas | 14 |
| 3.2. | Planificación temporal | 15 |
| 3.3. | Estimación de costes | 17 |
| 4. | Análisis de requisitos | 19 |
| 4.1. | Documentación de casos de uso | 19 |
| 4.2. | Requisitos de datos | 23 |
| 4.3. | Mapa del sitio | 25 |
| 5. | Desarrollo e integrado | 26 |
| 5.1. | Desarrollo de la aplicación | 26 |
| 5.1.1. | Modelos | 26 |
| 5.1.2. | Controladores | 27 |
| 5.1.3. | Vistas | 29 |
| 5.2. | Interfaz de la aplicación | 30 |
| 5.3. | Desarrollo del integrado | 35 |
| 5.3.1. | Jenkins | 35 |
| 5.3.2. | Enlazar Jenkins con BitBucket | 36 |
| 5.3.3. | Realizar los tests automáticamente | 36 |
| 5.3.4. | Enlazar Jenkins con Slack | 36 |
| 5.3.5. | Crear un sitio ftp y enlazarlo con Jenkins | 37 |
| 6. | Conclusiones y posibles mejoras | 38 |
| 6.1. | Conclusiones de los objetivos | 38 |
| 6.2. | Conclusiones personales | 38 |
| 6.3. | Posibles mejoras | 38 |
| 7. | Anexos | 40 |
| 7.1. | Anexo I (Configurar la base de datos en ASP .NET MVC 5) | 40 |
| 7.2. | Anexo II (Tutorial para la configuración de Jenkins) | 41 |
| 7.3. | Anexo III (Crear un sitio ftp en IIS y enlazarlo a Jenkins) | 53 |
| 8. | Bibliografía | 60 |

Índice de figuras:

| | |
|---|----|
| Figura 1: Mapa del sitio | 25 |
| Figura 2: Menú principal | 31 |
| Figura 3: Menú principal con varios xpaths asociados | 31 |
| Figura 4: Añadir un enlace | 32 |
| Figura 5: Añadir un xpath | 33 |
| Figura 6: Borrar un enlace | 33 |
| Figura 7: Borrar un xpath | 34 |
| Figura 8: Listar los elementos | 34 |
| Figura 9: Integrado con Jenkins | 35 |
| Figura 10: Descargar Jenkins | 41 |
| Figura 11: Gestionar Jenkins | 41 |
| Figura 12: Buscar el plugin de BitBucket | 42 |
| Figura 13: Plugin de BitBucket | 42 |
| Figura 14: Crear un nuevo proyecto en Jenkins | 42 |
| Figura 15: Configurar BitBucket en Jenkins 1 | 42 |
| Figura 16: Configurar BitBucket en Jenkins 2 | 43 |
| Figura 17: Configurar BitBucket en Jenkins 3 | 43 |
| Figura 18: Configurar BitBucket en Jenkins 4 | 44 |
| Figura 19: Estado de Jenkins una vez configurado | 44 |
| Figura 20: Plugin de MSTestRunner | 45 |
| Figura 21: Configurar MSTest 1 | 45 |
| Figura 22: Configurar MSTest 2 | 46 |
| Figura 23: Configurar MSTest 3 | 46 |
| Figura 24: Configurar test con Windows command batch | 47 |
| Figura 25: Configurar Slack 1 | 48 |
| Figura 26: Configurar Slack 2 | 48 |
| Figura 27: Configurar Slack 3 | 49 |
| Figura 28: Configurar Slack 4 | 49 |
| Figura 29: Configurar Slack 5 | 49 |
| Figura 30: Configurar Slack 6 | 50 |
| Figura 31: Enlazar Jenkins con Slack 1 | 50 |
| Figura 32: Enlazar Jenkins con Slack 2 | 50 |
| Figura 33: Enlazar Jenkins con Slack 3 | 51 |
| Figura 34: Enlazar Jenkins con Slack 4 | 51 |
| Figura 35: Enlazar Jenkins con Slack 5 | 52 |
| Figura 36: Ejemplo de funcionamiento del bot de Slack | 52 |
| Figura 37: Crear un sitio ftp 1 | 53 |
| Figura 38: Crear un sitio ftp 2 | 54 |
| Figura 39: Crear un sitio ftp 3 | 54 |
| Figura 40: Crear un sitio ftp 4 | 55 |

| | |
|---|----|
| Figura 41: Crear un sitio ftp 5 | 55 |
| Figura 42: Crear un sitio ftp 6 | 56 |
| Figura 43: Crear un sitio ftp 7 | 56 |
| Figura 44: Enlazar Jenkins al sitio ftp 1 | 57 |
| Figura 45: Enlazar Jenkins al sitio ftp 2 | 57 |
| Figura 46: Enlazar Jenkins al sitio ftp 3 | 57 |
| Figura 47: Enlazar Jenkins al sitio ftp 4 | 58 |
| Figura 48: Enlazar Jenkins al sitio ftp 5 | 58 |
| Figura 49: Enlazar Jenkins al sitio ftp 6 | 59 |
| Figura 50: Ejemplo 1 | 59 |

1. Introducción:

En este documento se explican y detallan cada uno de los componentes de EForce Scraper. Para entenderlo mejor se ofrece una descripción detallada de todo el comportamiento de la aplicación EForce Scraper, se presenta la motivación del proyecto y los objetivos del mismo.

1.1 Introducción al proyecto (EForce Scraper):

Para explicar que es un scraper, primero voy a intentar definir qué significa la palabra “scraper” en el mundo de la informática. Un scraper o web scraper, depende como quieras llamarlo, en el mundo informático, es un software desarrollado para poder extraer información de distintas páginas web.

Su objetivo es extraer información, habitualmente en formato HTML y procesarla para poder ser guardada en bases de datos, hojas de excel y/o ser presentada al usuario de una manera clara y bien estructurada.

La técnica del web scraping se viene utilizando cada vez más, ya que muchas empresas se han dado cuenta de las ventajas que proporciona el hecho de tener grandes cantidades de información actualizada a su disposición.

Esto proporciona la posibilidad de tomar decisiones en tiempo real.

Como en cualquier scraper, en este proyecto se desarrolla una aplicación web encargada de extraer información de otras páginas y organizar dicha información en una sola ventana (un scraper).. El usuario puede añadir enlaces a páginas web y la aplicación se encarga de guardar dichos datos en una base de datos. Por cada enlace se pueden guardar varios XPath (o ninguno). Los XPath pueden ser de tipo URL o Texto. Si un XPath es de tipo URL, se le puede añadir a su vez su propio XPath y así de forma sucesiva.

La aplicación, por cada XPath de tipo Texto, busca el enlace a la que pertenece y con ese enlace y XPath, extraer el nombre del producto y su valor (precio, temperatura etc.).

La aplicación guarda en la base de datos los enlaces y los XPath asociados, pero no guarda los nombres y los valores de los productos. No los guarda porque dicha información puede variar con el tiempo e interesa que el usuario tenga una información actualizada.

Por otra parte, el equipo de desarrollo puede comprobar el funcionamiento correcto de la aplicación mediante la incorporación de Slack y Jenkins. Jenkins está programado para bajarse periódicamente el proyecto desde la plataforma BitBucket y realizar los tests programados.

También la plataforma de comunicación Slack está integrada. Esta se ha programado para que avise cuando falla algún test o cuando el proyecto vuelve a la normalidad (no tiene fallos después de un periodo de fallos).

Además, se ha configurado una página ftp donde Jenkins va subiendo el proyecto, si este no tiene fallos.

1.2 Motivación:

Este proyecto se desarrolla porque la empresa EForce ICT desea utilizar las funciones de scraping para dar una ventaja competitiva a sus clientes. En este proyecto desea agilizar la manera de percibir la información de los usuarios, ya que permite organizar información desde diferentes fuentes en una sola página.

Esto permite a empresas clientes de EForce ICT poder monitorizar la competencia, sabiendo en tiempo real los precios de los productos de sus páginas web.

Además, la parte desarrollada de integración, ayuda a comprobar automáticamente si los cambios realizados en el proyecto causan algún problema, causan que los test fallen. Jenkins ejecuta periódicamente los tests y si ve que algún test falla, avisa a los trabajadores de EForce Spain de qué proyecto informático (proyecto o aplicación desarrollado por la empresa EForce) ha fallado mediante la aplicación Slack.

También facilita mucho el trabajo de la empresa el tener el proyecto subido automáticamente en un servidor ftp.

1.3 Objetivos:

El objetivo principal del proyecto es desarrollar un portal web capaz de extraer información desde diferentes páginas web y agruparla en una sola página, pero también integrar diferentes aplicaciones en el proyecto (Jenkins, Slack).

El objetivo principal se puede separar en dos partes, la parte del desarrollo del web scraper y la parte de integración.

La parte del desarrollo del web scraper se puede desglosar en los siguientes objetivos menores:

- Aprender a utilizar el framework ASP .NET MVC
- Utilizar HTML Agility Pack para extraer la información desde una página web
- Realizar la conexión del programa principal con la base de datos de Microsoft SQL Server

- Crear los modelos
- Crear los controladores para cada tipo de acción (añadir dominio/xpath, borrar dominio/xpath, listar productos)
- Crear las vistas con Razor
- Utilizar Bootstrap para mejorar la apariencia de la aplicación
- Comprobar que la aplicación funciona correctamente

La parte de la integración se puede desglosar en los siguientes objetivos menores:

- Aprender a instalar y configurar Jenkins
- Conseguir que Jenkins descargue el proyecto desde BitBucket
- Conseguir realizar tests automatizados en Jenkins con MS Visual Studio
- Configurar Jenkins para que ejecute los tests periódicamente
- Enlazar Slack con Jenkins
- Conseguir que Slack avise si falla algún test
- Conseguir que Slack avise si el proyecto vuelve a la normalidad (ya no hay ningún fallo en los tests)

2. Descripción del proyecto:

En este apartado se presenta la empresa EForce Spain, se describe el alcance del proyecto y las tecnologías utilizadas para desarrollar el mismo.

2.1 La empresa:

EForce Spain es una empresa dedicada al desarrollo de software, ubicada en Espaitec I de la Universidad Jaume I de Castellón. La empresa utiliza tecnologías como .NET y Umbraco CMS para desarrollar soluciones web para clientes, generalmente procedentes del extranjero.

La empresa desarrolla páginas web a medida para sus clientes, utilizando el framework ASP .NET, Microsoft SQL Server y el gestor de contenidos Umbraco. Con estas tecnologías la empresa es capaz de realizar páginas web integradas, eficientes y actuales.

La empresa está formada por un equipo de tres personas: un desarrollador *full-stack*, un desarrollador *front-end* y un diseñador.

2.2 Alcance:

El alcance de este proyecto consiste en desarrollar la aplicación en dos partes fundamentales. La aplicación y el integrado.

Por parte de la aplicación, esta debe permitir al usuario:

- Añadir un enlace
- Asociar un XPath a un enlace
- Borrar un XPath
- Borrar un enlace y todos los XPath asociados
- Listar los productos (nombre y valor de cada XPath asociado)

Por parte de la integración, esta debe permitir:

- Subir el proyecto a la página web
- Actualizar dicho proyecto
- Realizar tests sobre el proyecto
- Subir el proyecto a un servidor ftp
- Avisar por slack si un proyecto tiene fallos
- Avisar por slack si un proyecto vuelve a la normalidad

2.3 Tecnologías utilizadas:

Estas tecnologías son las utilizadas por la empresa EForce Spain, la empresa donde se hizo la estancia en prácticas y por lo tanto fueron las tecnologías utilizadas en el desarrollo de este proyecto final de grado.



ASP .NET MVC:

ASP .NET es un framework diseñado para páginas web dinámicas, aplicaciones web y otros servicios web. Desarrollado por Microsoft, el framework salió a la luz en 2002 como sucesor de Active Server Page (ASP).

ASP .NET MVC es una extensión de ASP .NET que permite crear páginas web utilizando el patrón de diseño modelo-vista-controlador (MVC).



C#:

C Sharp es un lenguaje de programación orientado a objetos. Ha sido desarrollado y Microsoft como parte de la plataforma .NET . Su sintaxis se basa en C y C++, aunque incluye el modelado orientado a objetos como Java y C++.

Aunque C# forma parte de .NET , en la actualidad se puede utilizar para desarrollar aplicaciones para otras plataformas, como Unix, Android, iOS, Mac OS y Linux.



Microsoft SQL Server:

Microsoft SQL Server es un sistema de manejo de bases de datos relacionales, desarrollado por Microsoft. Se puede utilizar con la consola de comandos, pero también con la interfaz gráfica Management Studio. La aplicación permite crear bases de datos y conectarlas con ASP .NET .



Internet Information Services (IIS):

Internet Information Services es un conjunto de servicios de Microsoft que permite transformar un ordenador / máquina virtual en un servidor web. IIS incluye los módulos para procesar las páginas desarrolladas en ASP .NET, pero también otro tipo de páginas, como las desarrolladas en Perl o PHP.



Slack:

Slack es una aplicación de comunicación en equipo multiplataforma (Windows, Mac OS, Linux, iOS, Android). La comunicación entre usuarios puede ser directa o mediante chats grupales (similar a las salas de chats). Los chats grupales pueden ser privados o públicos. Dicha aplicación permite incorporar todo tipo de servicios de terceros a su plataforma, como Google Drive, Dropbox, Skype, GitHub y BitBucket.



Jenkins

Jenkins:

Jenkins es un software de integración continua para el desarrollo del software. La aplicación fue desarrollada utilizando el lenguaje Java y es de software libre. La aplicación sirve para poder conectar proyectos en la nube (tipo GitHub o BitBucket) y realizar tests unitarios automatizados sobre los proyectos.



BitBucket:

BitBucket es un servicio de almacenamiento en la nube para proyectos informáticos. Esta permite a varios usuarios hacer modificaciones sobre el código y la estructura del proyecto, creando varias versiones del mismo si hiciese falta. BitBucket está desarrollado en Python con el framework Django.



Bootstrap:

Twitter Bootstrap es un framework para el front-end de código abierto. Está basado en HTML, CSS y JavaScript y contiene plantillas con formularios, menús, botones y otros elementos para el diseño de una página web.

HTML Agility Pack:

HTML Agility Pack es una librería para Microsoft Visual Studio que permite extraer información desde páginas web usando XPath o XSLT (Extensible Stylesheet Language Transformations).

3. Planificación:

En este apartado se expondrán las tareas desarrolladas en el proyecto, la planificación mediante el diagrama de Gantt y por último se intentará hacer una estimación de los costes económicos del proyecto.

3.1 Definición de tareas:

Las tareas han sido divididas en módulos y tareas para una mejor organización de los mismos. A continuación se presentan los módulos:

Inicio:

En este módulo se ha definido el proyecto junto al tutor y al supervisor de la empresa. Además se han definido la metodología del trabajo, la tecnología a emplear y la documentación necesaria.

Formación:

En este apartado se busca aprender de manera eficiente cómo trabajar con ASP .NET MVC. En concreto, familiarizarse con el lenguaje C#, los modelos, controladores y las vistas Razor.

Documentación y planificación del proyecto:

Este módulo se podría separar en dos submódulos. El primero consistiría en la planificación del proyecto: realizar el diagrama de Gantt, estimar los costes y realizar la propuesta técnica. El segundo submódulo consistiría en estudiar las páginas web de las que la aplicación extrae información.

Desarrollo del producto:

En este módulo estaba planeada la programación del back-end del proyecto en ASP .NET. Aunque está separado en el diagrama de Gantt, al este módulo ha sido realizado casi de manera concurrente con el siguiente módulo.

Desarrollo de la página web:

En este módulo estaba planeada la programación del front-end del proyecto mediante las vistas Razor de ASP .NET. Aunque aparece de manera separada en la planificación inicial en el diagrama de Gantt, al final este módulo ha sido realizado casi de manera concurrente con el anterior módulo.

Puesta en marcha:

En este apartado se tiene en cuenta la implementación de la aplicación en la empresa.

Documentación y presentación del trabajo final de grado:

En este apartado se tiene en cuenta tanto la redacción del proyecto, como la presentación del mismo y la redacción de los informes quincenal de la estancia en prácticas en la empresa.

También se acordó una reunión diaria, a primera hora de la mañana entre el supervisor y el alumno, donde el alumno informa al supervisor sobre el avance de su proyecto y el supervisor propone posibles mejoras o modificaciones al mismo

3.2 Planificación temporal:

La planificación se muestra con una estructura en cascada. En el diagrama de Gantt podemos observar una tabla y una gráfica. En la tabla se pueden ver los diferentes módulos planificados y sus tareas. Al lado de cada tarea se puede observar una estimación de su duración en horas. Al lado de cada módulo se puede ver la estimación para ese módulo, que es básicamente la suma de todas las horas de las tareas que pertenecen a dicho módulo.

La estimación total de las prácticas ha sido un total de 300 horas, sin tener en cuenta los informes bisemanales y la documentación del Trabajo de Fin de Grado. Aunque en el diagrama de Gantt aparece que las prácticas acaban el día 31 de mayo, en realidad las prácticas empezaron el 16 de marzo y acabaron el día 7 de junio. Esta pequeña diferencia se debe a algunos días de fiesta de Pascua, que no se han contabilizado en el Microsoft Project, herramienta utilizada para realizar el diagrama de Gantt.

3.3 Estimación de costes:

Después de presentar la planificación temporal, ahora se presenta al detalle la estimación de los costes.

Para dicha estimación, se considera el salario medio de programador de 1.500 € mensuales (fuente: google). Teniendo en cuenta que el proyecto ha durado 3 meses, podemos aproximar el coste de programar la aplicación en 4.500 €.

También hay que tener en cuenta el servidor ftp donde Jenkins subiría el proyecto y el ordenador en el que Jenkins se mantendrá activo. Si tenemos en cuenta el precio medio de un servidor en Azure, el coste mensual sería de 15 €, que anualmente ascendería a 180 € mensuales. Un ordenador de gama media, para mantener Jenkins activo, costaría alrededor de 600 €.

Desglose final de los costes:

| Recurso | Tiempo | Precio | Total |
|-------------|----------|------------------------|----------------|
| Programador | 3 meses | 1.500 € / mes | 4.500 € |
| Servidor | 12 meses | 15 € / mes | 180 € |
| Ordenador | - | 600 € | 600 € |
| | | Cantidad total: | 5.280 € |

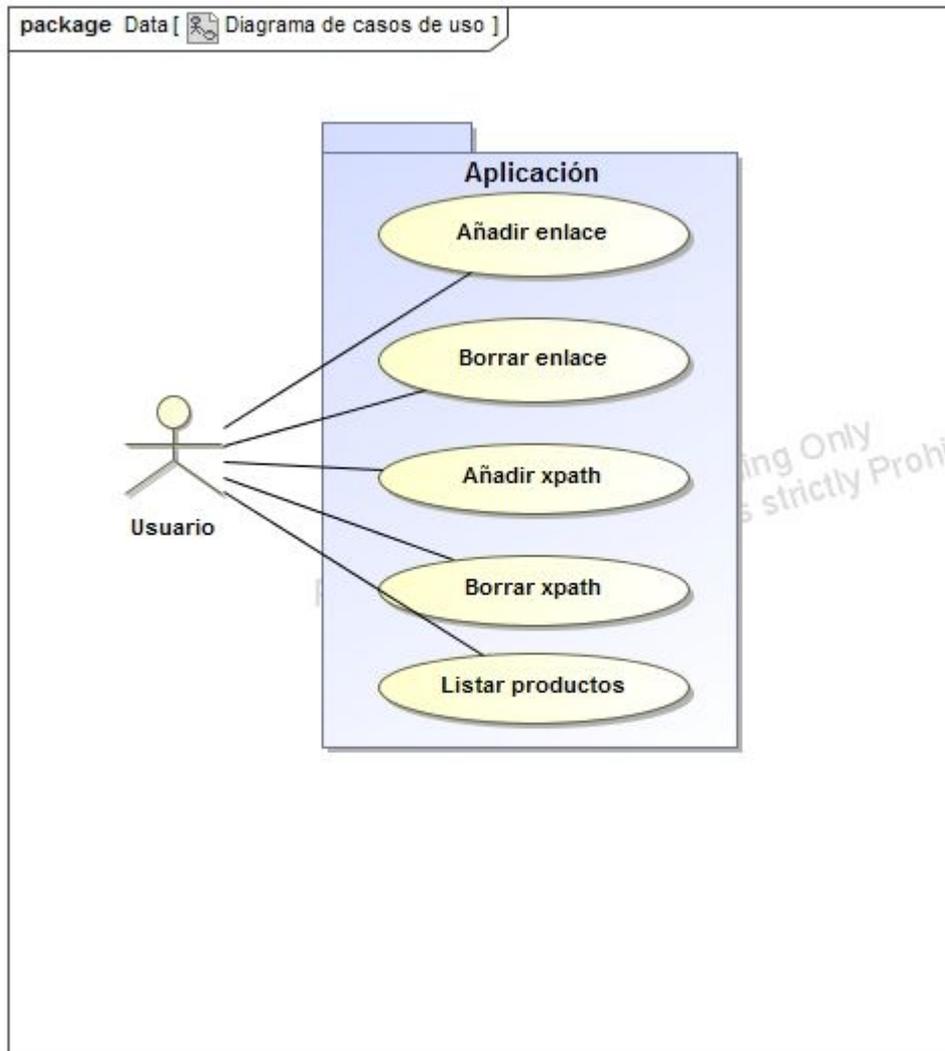
Por lo tanto, una vez detallados todos los recursos utilizados, el coste total estimado ascendería a 5.280 €.

4. Análisis de requisitos:

En este apartado se documentan los casos de uso y los requisitos de datos.

4.1 Documentación de casos de uso:

Casos de uso:



1.1. Caso de uso: Añadir enlace

| | | | |
|-------------------------------|---|-----------|------|
| Nombre del caso de uso | Añadir enlace | ID | CU01 |
| Complejidad | Complejidad baja | | |
| Descripción | Permite al usuario añadir un enlace. La aplicación guarda la el enlace en la base de datos y muestra el enlace en la página "Home". También la aplicación debe añadir al enlace las opciones de añadir xpath al enlace y borrar enlace. | | |
| Actores | · Usuario | | |
| Objetivos | Añadir un enlace a la aplicación | | |
| Pre Condición | | | |
| Post Condición | | | |

| Escenario | |
|------------------------------------|---|
| Flujo básico de los eventos | <p>Secuencia normal:</p> <ol style="list-style-type: none"> 1. La aplicación carga la página principal 2. El usuario selecciona "Añadir enlace" en la página principal 3. La aplicación carga el formulario para añadir un enlace 4. El usuario introduce el enlace 5. El usuario clicla el botón "Añadir enlace" 6. La aplicación guarda el enlace en la base de datos 7. La aplicación redirige el usuario a la página inicial 8. La aplicación muestra al usuario el nuevo enlace |

1.2. Caso de uso: Borrar enlace

| | | | |
|-------------------------------|--|-----------|------|
| Nombre del caso de uso | Borrar enlace | ID | CU02 |
| Complejidad | Complejidad baja | | |
| Descripción | Permite al usuario borrar un enlace. El sistema primero comprobará si hay algún xpath asociado al enlace, borrará los xpaths asociados si los hay y borrará el enlace de la base de datos. El enlace deberá desaparecer en la pantalla "Home". | | |
| Actores | · Usuario | | |
| Objetivos | Borrar un enlace de la aplicación | | |

| | |
|-----------------------|--|
| Pre Condición | |
| Post Condición | |

| Escenario | |
|------------------------------------|---|
| Flujo básico de los eventos | <p>Secuencia normal:</p> <ol style="list-style-type: none"> 1. La aplicación carga la página principal 2. El usuario selecciona “Borrar enlace” en el enlace deseado. 3. La aplicación carga el formulario para borrar un enlace 4. El usuario rellena el formulario 5. El usuario confirma clicando en “Borrar enlace” 6.1 La aplicación borra los xpath asociados de la base de datos 6.2 La aplicación borra el enlace de la base de datos 7. La aplicación redirige el usuario a la página inicial |

1.3. Caso de uso: Añadir xpath

| | | | |
|-------------------------------|--|-----------|------|
| Nombre del caso de uso | Añadir xpath | ID | CU03 |
| Complejidad | Complejidad baja | | |
| Descripción | Permite al usuario añadir un xpath a un enlace o a un xpath de tipo “URL”. La aplicación guarda el xpath en la base de datos y lo muestra en la página “Home”, debajo del enlace al que pertenece o xpath al que pertenece. Si el tipo de xpath es “Texto”, la aplicación debe añadir el botón “Borrar XPath” en la página principal, en cambio si el tipo de xpath es “URL”, la aplicación debe añadir los botones “Ver XPath asociado”, “Añadir XPath” y “Borrar XPath”. | | |
| Actores | · Usuario | | |
| Objetivos | Añadir un xpath a un enlace u otro xpath dentro de la aplicación | | |
| Pre Condición | | | |
| Post Condición | | | |

| Escenario | |
|------------------------------------|--|
| Flujo básico de los eventos | Secuencia normal: <ol style="list-style-type: none"> 1. La aplicación carga la página principal 2. El usuario selecciona "Añadir xpath" en el enlace o xpath deseado. 3. La aplicación carga el formulario para añadir un xpath 4. El usuario rellena el formulario 5. El usuario clicla el botón "Añadir XPath" 6. La aplicación guarda el xpath en la base de datos 7. La aplicación redirige el usuario a la página inicial 8. La aplicación muestra al usuario el xpath asociado al enlace o xpath deseado. |

1.4. Caso de uso: Borrar xpath

| | | | |
|-------------------------------|--|-----------|------|
| Nombre del caso de uso | Borrar xpath | ID | CU04 |
| Complejidad | Complejidad baja | | |
| Descripción | Permite al usuario borra un xpath. La aplicación buscará en la base de datos si hay otros xpaths asociados a él, borrará dichos xpath y borrará el xpath deseado de la base de datos y mostrará la página "Home" sin dicho xpath (y xpaths asociados). | | |
| Actores | · Usuario | | |
| Objetivos | Borrar un xpath de la aplicación | | |
| Pre Condición | | | |
| Post Condición | | | |

| Escenario | |
|------------------------------------|---|
| Flujo básico de los eventos | Secuencia normal: <ol style="list-style-type: none"> 1. La aplicación carga la página principal 2. El usuario selecciona "Borrar xpath" en el xpath deseado. 3. La aplicación carga el formulario para borrar un xpath 4. El usuario confirma clicando en "Borrar el xpath" 5.1 La aplicación borra los xpath asociados de la base de datos 5.2 La aplicación borra el xpath de la base de datos 6. La aplicación redirige el usuario a la página inicial |

1.5. Caso de uso: Listar productos

| | | | |
|-------------------------------|---|-----------|------|
| Nombre del caso de uso | Listar productos | ID | CU05 |
| Complejidad | Complejidad baja | | |
| Descripción | Permite al usuario listar los productos y sus valores (precios), además de un hipervínculo a cada producto. | | |
| Actores | · Usuario | | |
| Objetivos | Listar los productos obtenidos | | |
| Pre Condición | | | |
| Post Condición | | | |

| Escenario | |
|------------------------------------|--|
| Flujo básico de los eventos | Secuencia normal: <ol style="list-style-type: none">1. La aplicación carga la página principal2. El usuario selecciona "Valores" en la página principal3. La aplicación extrae los nombres, los valores y los enlaces de los xpath4. La aplicación muestra los nombres, valores y enlaces de los productos, agrupados por producto |

4.2 Requisitos de datos:

Requisitos de datos

ID: RD01

Nombre: Enlace

Autor: Dragos Verdes

Fuente: Jose Antonio Cano

Fecha creación: 10/04/2016

Fecha revisión: 12/04/2016

Fecha aceptación: 12/04/2016

Versión: 1.2

Tipo: Información

Datos a guardar: el enlace

Comentarios:

Requisitos de datos

ID: RD02

Nombre: Xpath

Autor: Dragos Verdes

Fuente: Jose Antonio Cano

Fecha creación: 10/04/2016

Fecha revisión: 12/04/2016

Fecha aceptación: 12/04/2016

Versión: 1.2

Tipo: Información

Datos a guardar: dirección, enlace padre, tipo, xpath al nombre, xpath al valor

Comentarios: Si el tipo del xpath es "URL", "xpath al valor" tendrá como valor un nulo.

4.3 Mapa del sitio:

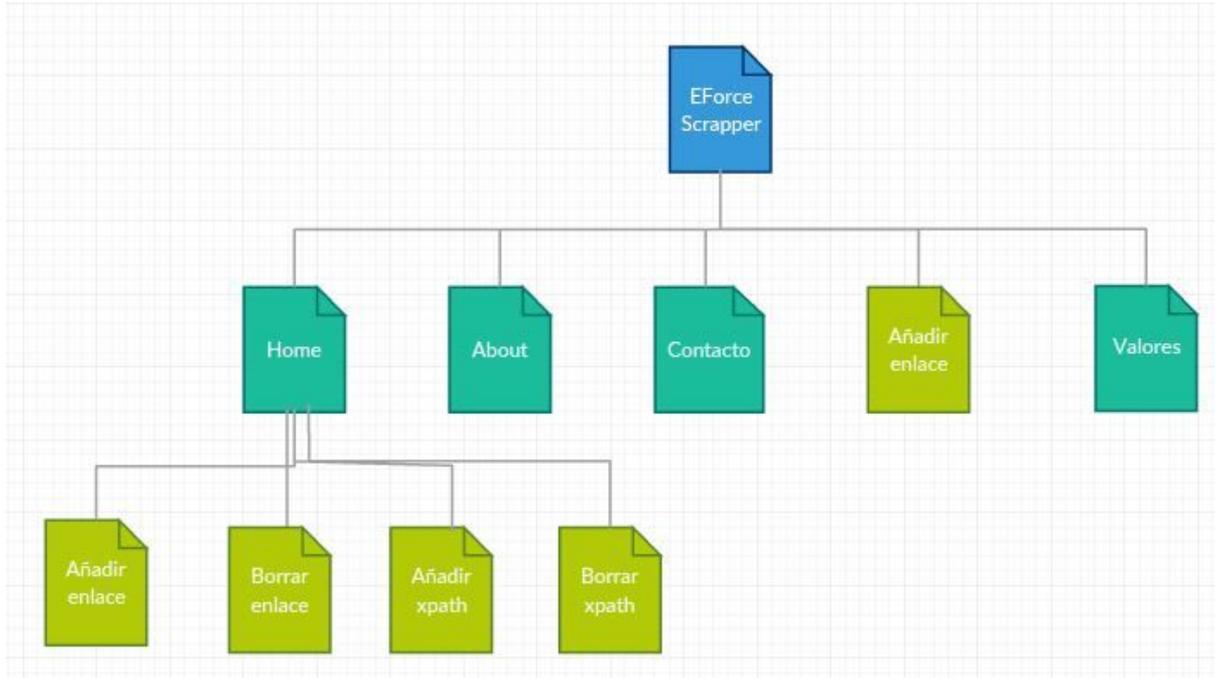


Figura 1 - Mapa del sitio

5. Desarrollo e integrado:

En este apartado se explica cómo se ha realizado el desarrollo de la aplicación y el integrado. El proyecto se puede separar en dos grandes bloques, la aplicación web y el integrado. Tanto la aplicación web, como el integrado con Jenkins pueden funcionar independientemente, por eso se explicarán los dos por separado.

5.1 Desarrollo de la aplicación:

Esta aplicación ha sido diseñada utilizando el patrón MVC (Modelo - Vista - Controlador). A continuación se explicarán los diferentes modelos, controladores y vistas utilizadas.

5.1.1 Modelos:

En este apartado se describen los modelos utilizados.

El primero ha sido el modelo “**Link**”, que contiene los siguientes atributos:

- Id (de tipo **long**): es un elemento “key”, que significa que es clave primaria en la base de datos. Es un número que sirve para identificar cada enlace o xpath.
- Direccion (de tipo **string**): sirve para guardar el enlace sobre el que se trabaja.
- LinkPadre (de tipo **long**): es un número que sirve para identificar el padre de un xpath. Sirve para saber sobre que enlace aplicar cada xpath.
- Tipo (de tipo **string**): sirve para saber si se trata de un enlace, un xpath de tipo texto o un xpath de tipo url.
- Nombre (de tipo **string**): es un texto que guarda el xpath al nombre de un producto.
- Valor (de tipo **string**): es un texto que guarda el xpath al valor de un producto.

El segundo modelo es “**LinkDB**”, que básicamente es una clase que sirve para crear la base de datos en Microsoft SQL Server.

El tercer modelo es “**Producto**”, que contiene los siguientes atributos:

- Nombre (de tipo **String**): sirve para almacenar el nombre del producto a mostrar por pantalla
- Valor (de tipo **String**): sirve para almacenar el valor del producto a mostrar por pantalla
- Enlace (de tipo **String**): sirve para almacenar el enlace donde se puede encontrar el producto al que se hace referencia

5.1.2 Controladores:

En este apartado se describen los siguientes controladores y sus “ActionResult”:

HomeController: Contiene la instancia a la base de datos.

Index: Extrae de la base de datos las direcciones y los xpath y llama a la vista con el mismo nombre.

ListProducts: Es el controlador encargado de extraer la información utilizando la librería HTML Agility Pack. Primero crea una lista de vacía de tipo **Producto**, luego se extrae toda la información de la base de datos y por cada elemento de tipo “texto” se extrae su nombre y su valor utilizando HTML Agility Pack. Si El nombre y el valor del producto no son nulos, se asigna el enlace y se guarda en la lista de tipo **Producto**. Finalmente se envía dicha lista a la vista con el mismo nombre que el controlador.

DeleteLink: Este controlador contiene dos métodos “ActionResult” llamadas “DeleteLink”. Uno es de tipo “HttpGet” y otro que es de tipo “HttpPost”.

El primero recibe un id de tipo long, busca en la base de datos dicho id y lo guarda en una variable de tipo link. Si dicha variable no es igual a null, se llama a la vista para generar el formulario de borrado.

El segundo método de tipo “ActionResult” recibe una variable de tipo link y busca en la base de datos todos los enlaces y xpath que tienen como padre o abuelo dicho valor y los borra. Luego busca el enlace pasado como parámetro y lo borra de la base de datos y después llama a la función “SaveChanges()” para guardar la base de datos.

DeleteXPath: Este controlador también contiene dos métodos “ActionResult” llamadas “DeleteXPath”. Uno es de tipo “HttpGet”, mientras el otro es de tipo “HttpPost”.

El método “ActionResult” de tipo “HttpGet” recibe un identificador de tipo long, busca en la base de datos dicho identificador y lo guarda en una variable de tipo link. Si dicha variable es diferente a null, se llama a la vista que genera el formulario de borrado de xpath.

El segundo método “ActionResult” recibe el enlace y busca en la base de datos para encontrar dicho enlace. Si lo encuentra, lo borra y llama a la función “SaveChanges()” para guardar la base de datos.

AddXPath: Este controlador también contiene dos métodos “ActionResult” con el mismo nombre del controlador, que también son de tipo “HttpGet” y “HttpPost”, respectivamente.

El primero recibe dos elementos de tipo string - la dirección del enlace y el id padre - y llama a la vista encargada de generar el formulario.

El segundo recibe una variable de tipo link y lo primero que hace es comprobar si el elemento link es de tipo “Texto” o “URL”:

- Si es de tipo “URL”, extrae de la variable la dirección. Con la dirección extrae el dominio de la página web. A ese dominio le añade el atributo link.Nombre y se crea la nueva dirección sobre cuál podrán aplicar nuevos xpath de tipo “Texto”. Después se crea un nuevo elemento link con el atributo dirección siendo la nueva dirección creada. Dicho elemento link se guarda en la base de datos y se llama a la función de guardado “SaveChanges()”.
- Si en cambio es de tipo “Texto”, se crea un nuevo elemento de tipo link y se añade a la base de datos. Después se llama a la función “SaveChanges()” que guarda la base de datos.

Link: Este controlador se encarga de añadir un enlace en la base de datos.

También contiene dos métodos de tipo “ActionResult”. Los dos métodos también son de tipo “HttpGet” y “HttpPost”, respectivamente.

El primero no recibe ningún parámetro, crea un elemento link vacío y llama a la vista encargada de generar el formulario.

El segundo método recibe un elemento de tipo link, guarda la dirección que contiene dicho elemento. Con esa dirección crea un nuevo elemento de tipo link y lo añade a la base de datos. Luego llama a la función “SaveChanges()” para guardar los cambios en la base de datos.

Nota: Todos los métodos de tipo “HttpPost” de los controladores, al acabar su tarea, redirigen al usuario a la página principal.

5.1.3 Vistas:

About:

About es una vista simple, encargada de mostrar información miscelánea de la aplicación.

AddXPath:

Es una vista a la que se llama desde el controlador “AddXPath”, desde el método de tipo “HttpGet”. Esta vista genera un formulario para que el usuario pueda añadir el tipo del xpath, el nombre y el valor. Una vez que el usuario envía los datos, se llama al controlador de tipo “HttpPost”.

Contact:

Contact es una vista simple, encargada de mostrar información de contacto con el desarrollador de la aplicación.

DeleteLink:

Es una vista que se llama desde el controlador “DeleteLink”, desde el método de tipo “HttpGet”. La vista genera una página donde se le muestra al usuario el enlace que desea borrar y le pide confirmación del borrado.

Si el usuario clica en “Borrar enlace”, se envía la información al método de tipo “HttpPost”, que se encarga de borrar el enlace y los xpaths asociados a él.

DeleteXPath:

Esta vista se llama desde el controlador “DeleteXPath”. Se genera una página donde se le muestra al usuario la información del xpath que desea borrar.

Si el usuario clica en “Borrar XPath”, se envía la información al método “ActionResult” de tipo “HttpPost”, que se encarga de borrar el xpath de la base de datos.

Index:

Es la vista encargada de generar la página principal. En ella se muestran todos los enlaces con los xpaths asociados.

Link:

Es una vista a la que se llama desde el controlador "Link". Sirve para generar un formulario para poder recoger un enlace y guardarlo en la base de datos. Una vez que el usuario introduce el enlace y clic en "Añadir enlace", la información se envía al controlador de tipo "HttpPost" para guardar el enlace en la base de datos.

ListProducts:

ListProducts es una vista que se encarga de mostrar los productos, sus valores y los enlaces a dichos productos. Es una vista que se llama desde el controlador "ListProducts".

5.2 Interfaz de la aplicación web:

A continuación se mostrará - con la ayuda de las capturas de pantalla - la interfaz gráfica de la aplicación web.

Esta aplicación ha sido diseñada para usuarios con un nivel informático medio o alto, por lo que se ha buscado agilizar las operaciones a cambio de un mayor riesgo de error. Como consecuencia, se permite la entrada de texto libre al rellenar el formulario de un xpath. Para evitar posibles errores, se pide confirmación cada vez que se desea borrar un dominio o xpath.

También para facilitar la navegación, toda pantalla tiene un botón para volver al menú principal y una barra de navegación de fácil acceso.

En la primera captura de pantalla se puede ver el menú principal. Al iniciar la aplicación se cargan los dominios, pero no se muestran los XPath asociados (figura 2).

Enlace: <https://www.hairworldshop.nl/product/shampoo/anti-roots/5410/instant-clear-pure/>

Ver XPath asociado
Añadir XPath
Borrar Dominio

Figura 2 - Menú principal

En cambio, en la segunda captura de pantalla se puede ver el menú principal, con los los xpaths desplegados. Se puede observar que si un xpath es de tipo URL, aparece la opción de “Añadir XPath”, en cambio si es de tipo texto, esa opción no aparece.

Enlace: <https://www.hairworldshop.nl/product/shampoo/anti-roots/5410/instant-clear-pure/>

Ver XPath asociado
Añadir XPath
Borrar Dominio

Enlace: <http://www.eltiempo.es/castellon-de-la-plana.html>

Ver XPath asociado

Id: 45
XPath al nombre: `//*[@id="page"]/main/div[4]/div/section[3]/section/div/article/div[1]/div[1]/header/h1`
XPath al valor: `//*[@id="page"]/main/div[4]/div/section[3]/section/div/article/div[1]/div[3]/div[1]/p`
Id del dominio padre: 44
Tipo: Texto
Borrar XPath

Id: 46
XPath al enlace: `//*[@id="page"]/main/div[4]/div/section[8]/article/ul/li[2]/a`
Id del dominio padre: 44
Tipo: URL

Ver XPath asociado
Añadir XPath
Borrar XPath

Añadir XPath
Borrar Dominio

Figura 3 - Menú principal con varios xpaths asociados

Al seleccionar “Añadir enlace”, el usuario puede añadir una dirección / un dominio.



The screenshot shows a web application interface with a dark navigation bar at the top containing the text 'EForce Scraper', 'Home', 'About', 'Contacto', 'Añadir enlace', and 'Valores'. Below the navigation bar, the page title is 'Añadir dominio'. Underneath the title, the word 'Dominio' is displayed. A horizontal line separates the header from the form area. The form contains a label 'Direccion' next to an empty text input field. Below the input field is a black button with the text 'Añade dominio' in white. To the left of the main form area, there is a separate button labeled 'Volver'.

Figura 4 - Añadir un enlace

Al seleccionar “Añadir XPath” en un dominio o xpath de tipo URL, se lleva al usuario al formulario de la figura 5. Allí el usuario puede añadir el tipo, el nombre y el valor.

Añadir XPath

Dirección

LinkPadre

Tipo

Nombre
XPath al nombre

Valor
XPath al valor (si es texto)

Añade XPath

[Volver](#)

Figura 5 - Añadir un xpath

Como se ha mencionado antes, a la hora de borrar un dominio se le presenta al usuario la dirección que desea borrar y una confirmación de dicha acción. Esto se ha implementado para evitar errores.

Borrar dominio

¿Estás seguro de que deseas borrar este dominio?

Dominio <https://www.hairworldshop.nl/product/shampoo/anti-roos/5410/instant-clear-pure/>

Borrar dominio | [Volver](#)

Figura 6 - Borrar un enlace

En el caso de borrar un xpath, se le presenta al usuario toda la información relacionada a dicho xpath (dirección, tipo, xpath al nombre y al valor).

EForce Scraper Home About Contacto Añadir enlace Valores

Borrar XPath

¿Estás seguro de que deseas borrar este XPath?

| | |
|-------------------------|--|
| Dirección | http://www.eltiempo.es/castellon-de-la-plana.html |
| Id del dominio | 44 |
| Tipo | Texto |
| XPath del nombre | //*[@id="page"]/main/div[4]/div/section[3]/section/div/article/div[1]/div[1]/header/h1 |
| XPath del valor | //*[@id="page"]/main/div[4]/div/section[3]/section/div/article/div[1]/div[3]/div[1]/p |

[Borrar XPath](#) | [Volver](#)

Figura 7 - Borrar un XPath

Por último se puede observar la página que lista los elementos. En dicha lista, por cada elemento se presentan el nombre y el valor de los xpaths asociados, además de la dirección.

EForce Scraper Home About Contacto Añadir enlace Valores

Listado de elementos

Nombre: Castellón de la Plana
Valor: 29°C
Enlace: [Visitar página](#)

Nombre: Almazora
Valor: 29°C
Enlace: [Visitar página](#)

[Volver](#)

Figura 8 - Listar los elementos

5.3 Desarrollo del integrado:

El integrado consiste en poder automatizar tareas en el desarrollo de un proyecto informático.

Los programadores de hoy en día suelen tener sus proyectos en la nube, en páginas como BitBucket o GitHub. Esto permite a los usuarios trabajar de manera concurrente y no perder información valiosa, además de poder volver a versiones anteriores de dichos proyectos.

Con el integrado desarrollado en este proyecto se pretende agilizar aún más el desarrollo de proyectos informáticos mediante la automatización de los tests, lo que permite a los programadores no tener que comprobar si un proyecto pasa los tests cada vez que se actualiza una clase o un método. También se ha integrado la posibilidad de subir ficheros de un proyecto a un servidor ftp, una vez que dichos ficheros hayan pasado los tests.

El integrado puede considerarse una entidad diferente a la aplicación web, ya que puede funcionar con otras aplicaciones y además, con otras tecnologías, no sólo con ASP .NET.

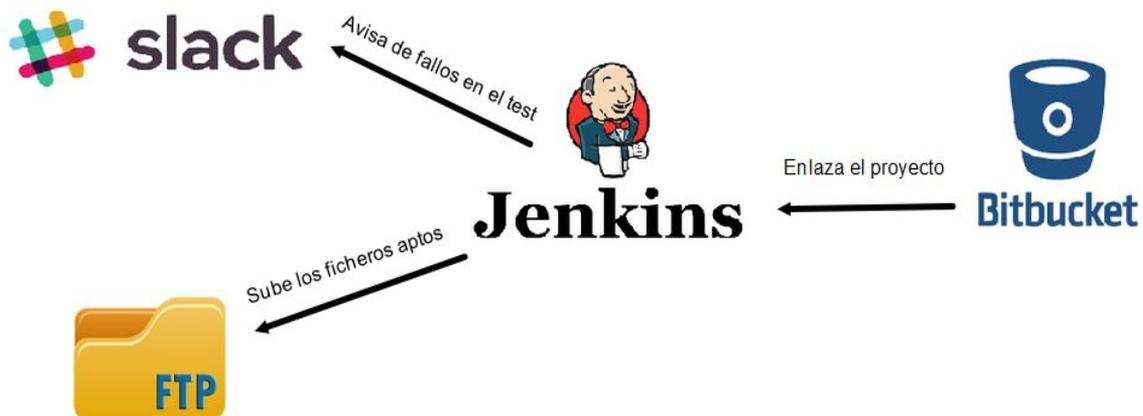


Figura 9 - Integrado con Jenkins

5.3.1 Jenkins:

Jenkins es la parte más importante del integrado. Jenkins se ejecuta como un servicio en un ordenador y permite descargar y automatizar tests sobre proyectos, para detectar rápido y automáticamente errores dentro del código de proyectos informáticos.

Además de su rapidez y automatización, Jenkins tiene multitud de plugins disponibles, lo que aumenta su versatilidad. En este proyecto he utilizado dos plugins: el de Slack para enviar notificaciones y el de MSTestRunner, para realizar los tests de manera automática.

En los siguientes apartados voy a comentar un poco por encima cómo la idea del integrado y cómo se unen las diferentes partes en un sólo integrado.

Los detalles de cómo instalar Jenkins y los pasos a seguir para integrar BitBucket, Slack y MSTestRunner para realizar tests automáticos están disponibles en el Anexo 2, donde se explica paso a paso cómo configurar todas las funciones.

En el Anexo 3 se encuentra explicado paso a paso cómo crear una página ftp y cómo enlazarlo a Jenkins para que suba ficheros automáticamente, una vez que pasen los tests.

5.3.2 Enlazar Jenkins con BitBucket:

El siguiente paso hacía el integrado de Jenkins en nuestro proyecto - y en cualquier otro proyecto - es conectar BitBucket con Jenkins. El objetivo es conseguir descargar el proyecto en el que se trabaja a una máquina en la que se esté ejecutando Jenkins.

5.3.3 Realizar los tests automáticamente:

Una vez descargado el proyecto, el siguiente paso es automatizar los tests. Jenkins realizará los tests que encuentre en el proyecto. En el anexo 2 se explican dos maneras para hacerlo. Una es con el plugin MSTestRunner y la otra manera es con el comando batch de Windows. La única condición es tener instalado Microsoft Visual Studio.

Jenkins permite dos maneras de programar cuándo ejecutar los tests. Una es mediante un webhook de BitBucket, cada vez que se hace un “push” en el proyecto, Jenkins ejecuta los tests. La otra manera de hacerlo es programarlo mediante una sintaxis similar a cron, en la que podemos especificar en qué día, a qué hora, minuto y segundo queremos realizar los tests. En este proyecto he optado por la segunda opción.

5.3.4 Enlazar Jenkins con Slack:

Una vez Jenkins descarga el proyecto y ejecuta los tests, se introduce la opción de enlazar Slack. Slack es una aplicación de comunicación en equipo multiplataforma. El objetivo es avisar a los componentes del equipo que trabaja en

un proyecto informático de si los tests fallan o de si el proyecto vuelve a la normalidad (ya no fallan los tests).

Para enlazarlo solamente nos hace falta instalar el plugin de Slack y configurar un bot en slack para que avise en los canales de comunicación deseados.

5.3.5 Crear un sitio ftp y enlazarlo a Jenkins:

También en la parte de la integración se ha creado una página ftp y se ha enlazado a Jenkins.

La página ftp ha sido creada en un servidor con IIS (Internet Information Service). Gracias a IIS se configura fácilmente un sitio ftp. Jenkins ya tiene integrada la opción de publicar archivos en páginas ftp.

La creación del sitio ftp y como configurarlo está disponible en el anexo 3.

6. Conclusiones y posibles mejoras:

6.1 Conclusiones de los objetivos:

Todos los objetivos del proyecto se han alcanzado de forma satisfactoria. La aplicación web almacena los enlaces y los XPath en la base de datos y los extrae de forma correcta. También extrae correctamente la información de las páginas web utilizando la librería HTML Agility Pack y consigue mostrar toda la información adecuadamente.

Por otra parte, Jenkins accede correctamente al proyecto mediante BitBucket y consigue realizar los tests y sube los ficheros de manera correcta a la página ftp. También Slack avisa correctamente los resultados de dichos tests.

6.2 Conclusiones personales:

Como una conclusión particular, estoy muy contento con el proyecto. Ha sido el primer proyecto que he desarrollado completamente sólo y desde cero. También ha sido una gran oportunidad para aprender nuevas tecnologías. Aunque me ha costado, he aprendido a utilizar el framework ASP .NET. Había trabajado antes con el modelo MVC, pero ha sido con el framework Spring. También he aprendido el lenguaje de programación C#, aunque este lenguaje me ha costado poco aprenderlo, ya que es muy parecido a Java. También ha sido la primera vez que he trabajado con Jenkins, que me ha parecido una herramienta muy útil para automatizar tests y también ha sido la primera vez que he creado una página ftp.

También he aprendido a buscar información y soluciones a errores en internet. De gran ayuda han sido los tutoriales y los manuales online, pero también las páginas de ayuda, como **stackoverflow.com** .

6.3 Posibles mejoras:

A falta de tiempo, no me ha dado tiempo a mejorar la aplicación. Entre las posibles mejoras me habría gustado mejorar los formularios de para evitar errores. También mejoraría el diseño de la página donde se listan los productos, para tenerlos mejor organizados, por dominio y de una forma más llamativa. Otra mejora sería a la hora de clicar en “Añadir XPath”, el formulario fuese dinámico y no tener que acceder a otra página.

También mejoraría el modelo MVC, donde aumentaría el número de modelos, uno para el enlace, otro para xpath de tipo texto y otro para xpath de tipo url.

Otra cosa que me se podría mejorar serían los formularios, especialmente a la hora de añadir un xpath. Si el xpath fuese de tipo "URL", se desactiva el campo "Valor".

ANEXOS:

Anexo I (Configurar la base de datos en ASP .NET MVC 5):

1. En el fichero **Web.config** más externo hay que añadir:

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data
  Source=(LocalDb)\MSSQLLocalDB;AttachDbFilename=|DataDirectory|\EForc
  eScrapper.mdf;Initial Catalog=EForceScrapper;Integrated Security=True"
  providerName="System.Data.SqlClient" />
</connectionStrings>
```

2. En el fichero **App.config**, se debe añadir:

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data
  Source=(LocalDb)\MSSQLLocalDB;AttachDbFilename=|DataDirectory|\EForc
  eScrapper.mdf;Initial Catalog=EForceScrapper;Integrated Security=True"
  providerName="System.Data.SqlClient" />
</connectionStrings>
```

3. En “**Developer Command Prompt for VS2015**” de Windows, se deben ejecutar los siguientes comandos:

```
sqllocaldb.exe stop MSSQLLocalDB
```

```
sqllocaldb.exe delete MSSQLLocalDB
```

```
sqllocaldb.exe start MSSQLLocalDB
```

4. Dentro de la clase **LinkDB**, añadir dentro del constructor

```
AppDomain.CurrentDomain.SetData("DataDirectory",
System.IO.Directory.GetCurrentDirectory());
```

Anexo II (Tutorial para la instalación de Jenkins, programar los tests y enlazar Slack):

Para instalar Jenkins vamos a su página web y elegimos en el listado la versión para nuestro sistema operativo:

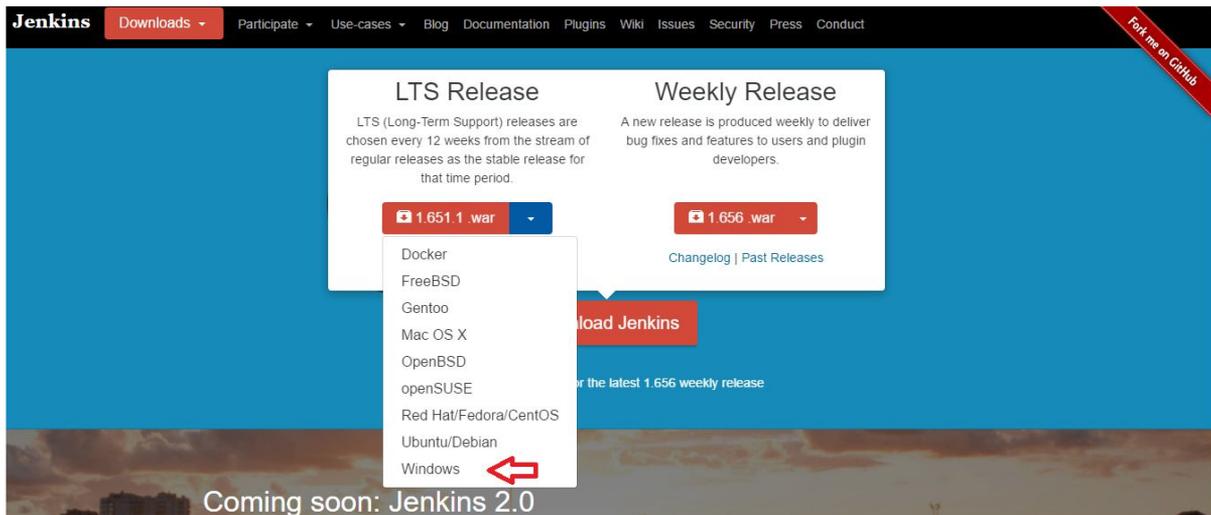


Figura 10 - Descargar Jenkins

Una vez instalado Jenkins, tenemos que añadir el plugin de Bitbucket. Para eso (dentro de Jenkins) vamos a “Manage Jenkins” - “Manage plugins”:



Figura 11 - Gestionar Jenkins

Dentro de “Manage plugins” seleccionamos la pestaña “Available” y buscamos “Bitbucket”:



Figura 12 - Buscar el plugin de BitBucket

Seleccionamos el plugin “Bitbucket Plugin” con la descripción “Integrates with BitBucket”:



Figura 13 - Plugging de BitBucket

Seleccionamos “Download now and install after restart”. Mientras se va instalado clicamos la casilla “Restart Jenkins when installation is complete and no jobs are running”.

Nota: Si se queda estancado después de la instalación, podemos volver a la pantalla de inicio de Jenkins.

Una vez instalado el plugin, clicamos en “New item” en el menú de la izquierda. Allí le damos un nombre al objeto y seleccionamos la opción “Freestyle” y seleccionamos “OK”:



Figura 14 - Crear un nuevo proyecto en BitBucket

En la nueva ventana podemos comprobar que el plugin de Bitbucket se ha instalado correctamente, en el apartado “Build Triggers” nos aparecerá la opción “Build when a change is pushed to BitBucket”:

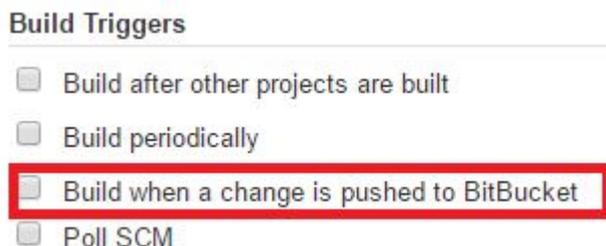


Figura 15 - Configurar BitBucket en Jenkins 1

Dentro de la misma pantalla tendremos que configurar la conexión a Bitbucket. Para ello, en el apartado “Source Code Management” seleccionamos la opción “Git”. Una vez seleccionada esa opción insertamos el enlace al repositorio de Bitbucket en el campo “Repository URL”:



The screenshot shows the 'Source Code Management' section of the Jenkins configuration page. Under the 'Repositories' heading, four radio buttons are listed: 'None', 'CVS', 'CVS Projectset', and 'Git'. The 'Git' option is selected. To the right, there is a text input field labeled 'Repository URL' containing the text 'https://usuario@bitbucket.org/nombre/proyecto.git'.

Figura 16 - Configurar BitBucket en Jenkins 2

Nos aparecerá un fallo en color rojo, no hay de qué preocuparse. En “Credentials” seleccionamos “Add” para poder conectarnos:



The screenshot shows the 'Credentials' section of the Jenkins configuration page. A dropdown menu is set to '- none -'. To the right of the dropdown is a button with a key icon and the text 'Add'. This button is highlighted with a red rectangular border.

Figura 17 - Configurar BitBucket en Jenkins 3

Al seleccionar “Add” se nos abre una ventana. En “Kind” seleccionamos “Username with password”, la opción “Scope” la dejamos sin modificar, en “Username” añadimos nuestro usuario y en “Password” la contraseña. “Description” es opcional. Después seleccionamos “Add”:

Add Credentials

Kind: Username with password

Scope: Global (Jenkins, nodes, items, all child items, etc)

Username: Usuario

Password:

Description:

Add Cancel

Figura 18 - Configurar BitBucket en Jenkins 4

Después de seleccionar “Add”, debería desaparecer el aviso de error. Al final de la página seleccionamos “Save”.

Si todo va bien deberíamos llegar a esta página:

Jenkins > my-bitbucket-project2 >

Back to Dashboard

Status

Changes

Workspace

Build Now

Delete Project

Configure

Project my-bitbucket-project2

Workspace

Recent Changes

Permalinks

Build History trend

find

RSS for all RSS for failures

Figura 19 - Estado de Jenkins una vez configurado

Para configurar los tests con MSTestRunner plugin:

Primero debemos instalar el plugin MSTestRunner:



Figura 20 - Plugin de MSTestRunner

Hay dos maneras de ir ejecutando los test, una con MSTest y otra con comandos batch de Windows.

Con MSTest:

En el menú principal, seleccionamos "Manage Jenkins" y una vez allí seleccionamos "Configure System". En el apartado "MSTest" seleccionamos "Add MSTest". Se nos abrirá un formulario, en el que debemos añadir el nombre, la ruta a "mstest.exe". También seleccionamos "Omit No Isolation". Seleccionamos "Apply" y después "Save":

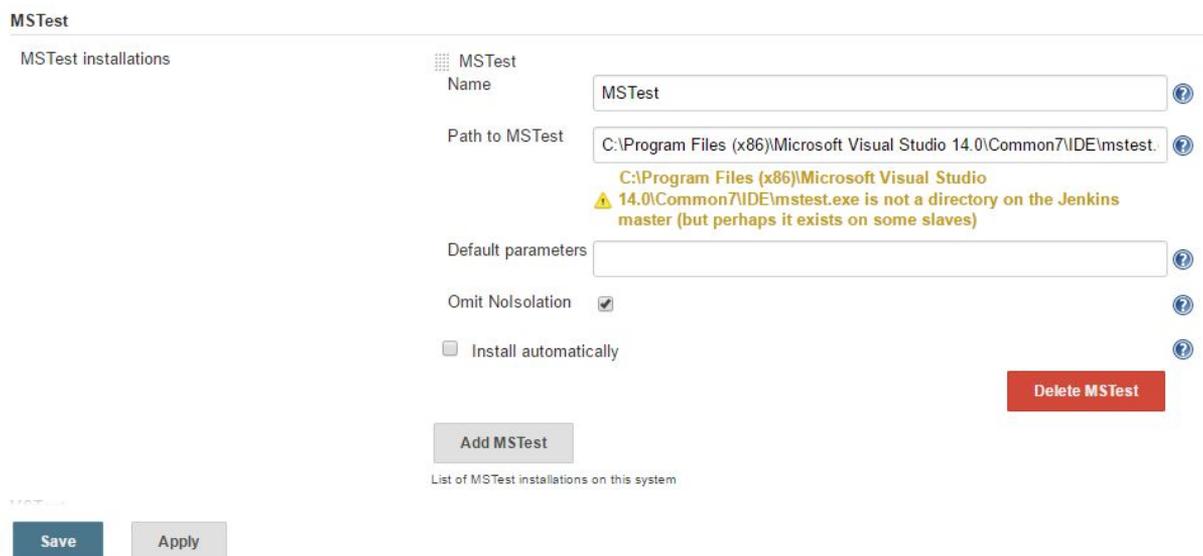


Figura 21 - Configurar MSTest 1

Después en entramos en nuestro proyecto y entramos en "Configure". En el apartado "Build", seleccionamos "Add build step" y elegimos la opción "Run unit tests with MSTest":

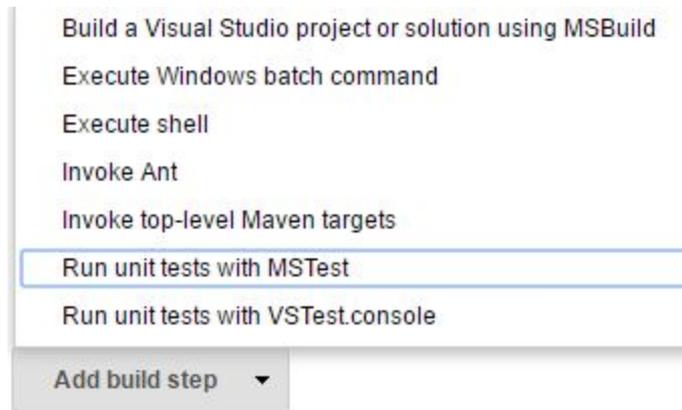


Figura 22 - Configurar MSTest 2

Se nos abrirá un formulario en el que tenemos que seleccionar el Test (MSTest lo hemos creado en el paso anterior), en “Test Files” ponemos la ruta al fichero test (.dll) y ponemos la ruta al fichero donde queremos guardar información de los test. También seleccionamos “Continue on failed tests”:

Figura 23 - Configurar MSTest 3

Después seleccionamos “Apply” y “Save”.

Con Windows batch command:

Vamos a la configuración de nuestro proyecto y allí, en el apartado “Build” seleccionamos “Add build step” y elegimos la opción “Execute Windows batch command”:

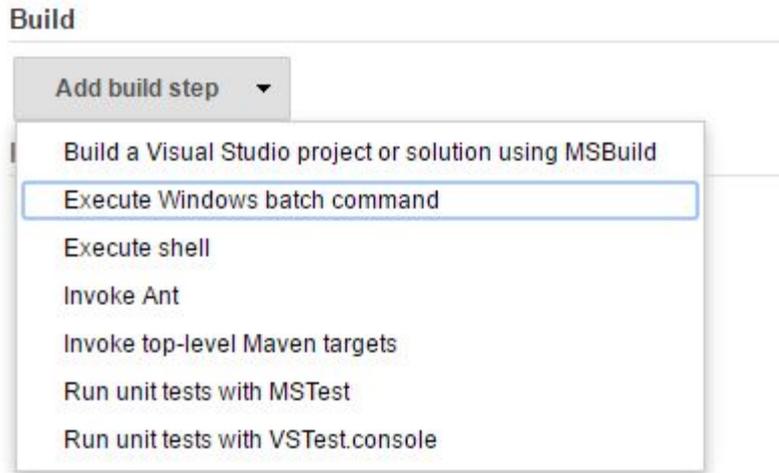


Figura 24 - Configurar test con Windows command batch

Una vez hecho eso, se nos abrirá un recuadro donde añadiremos la siguiente sentencia:

```
"C:\Program Files (x86)\Microsoft Visual Studio 14.0\Common7\IDE\mstest.exe"  
/testcontainer:"C:\Program Files  
(x86)\Jenkins\jobs\my-bitbucket-Dragos\workspace\EForceScrapper.Tests\bin\Deb  
ug\EForceScrapper.Tests.dll"  
/resultfile:"C:\Program Files  
(x86)\Jenkins\jobs\my-bitbucket-Dragos\workspace\Resultado_%BUILD_NUMBE  
R%.txt"
```

En la primera línea, entre comillas, debemos introducir la ruta a "mstest.exe", que se encuentra en la carpeta donde está instalado el Microsoft Visual Studio.

En la segunda línea, después de "/testcontainer:" debemos introducir entre comillas la ruta al fichero de tests (.dll) de nuestro proyecto.

En la tercera línea, después de "/resultfile:", debemos introducir entre comillas el fichero donde se guardará información sobre el test. Al nombre del fichero resultante, deberíamos añadir "%BUILD_NUMBER%", ya que si el fichero existe, nos dará un fallo de ejecución.

Las líneas deben ser separadas por un espacio en blanco y no por salto de carro. Después seleccionamos "Apply" y después "Save".

Ya podemos ejecutar el build.

Para incorporar Jenkins a Slack

En la parte de Slack:

Entramos en nuestro canal y elegimos la opción “Settings” y en el desplegable que aparece seleccionamos “Add an app or integration”:

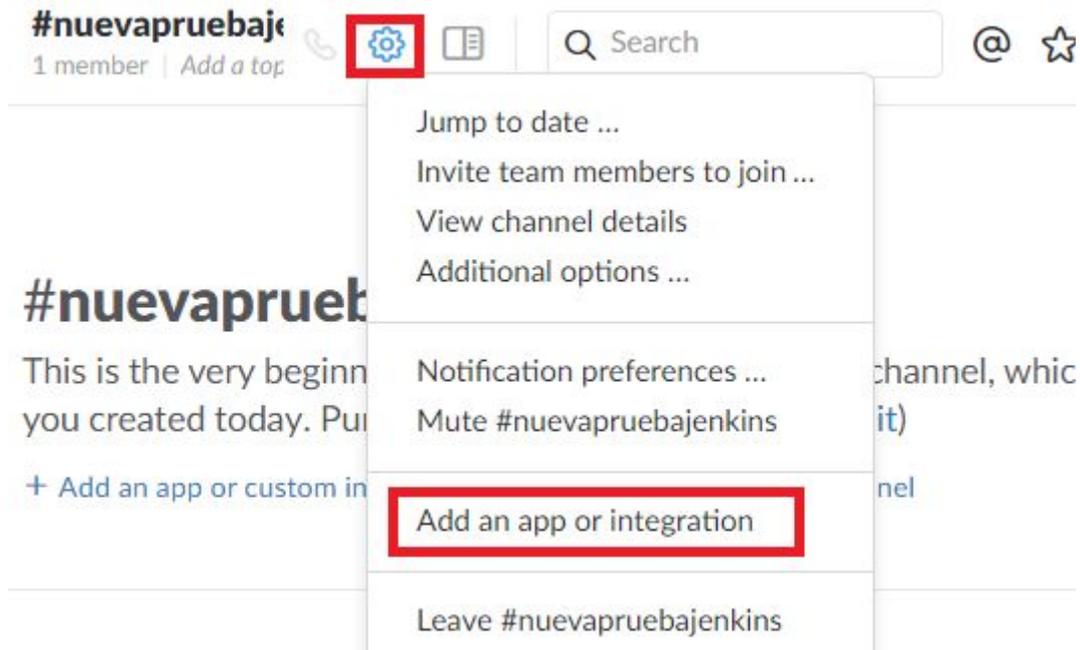


Figura 25 - Configurar Slack 1

Esto nos llevará a la página web de Slack, donde nos pedirá que nos identifiquemos. Una vez identificados, nos aparecerá un campo de búsqueda, donde introducimos “Jenkins”. Seleccionamos “Jenkins CI”:

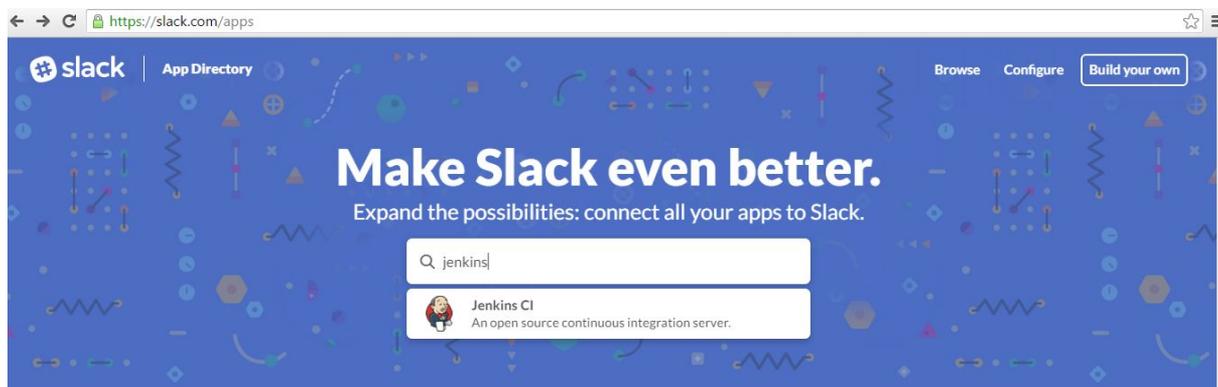


Figura 26 - Configurar Slack 2

En la siguiente página nos pedirá elegir en qué equipo queremos integrar Jenkins:

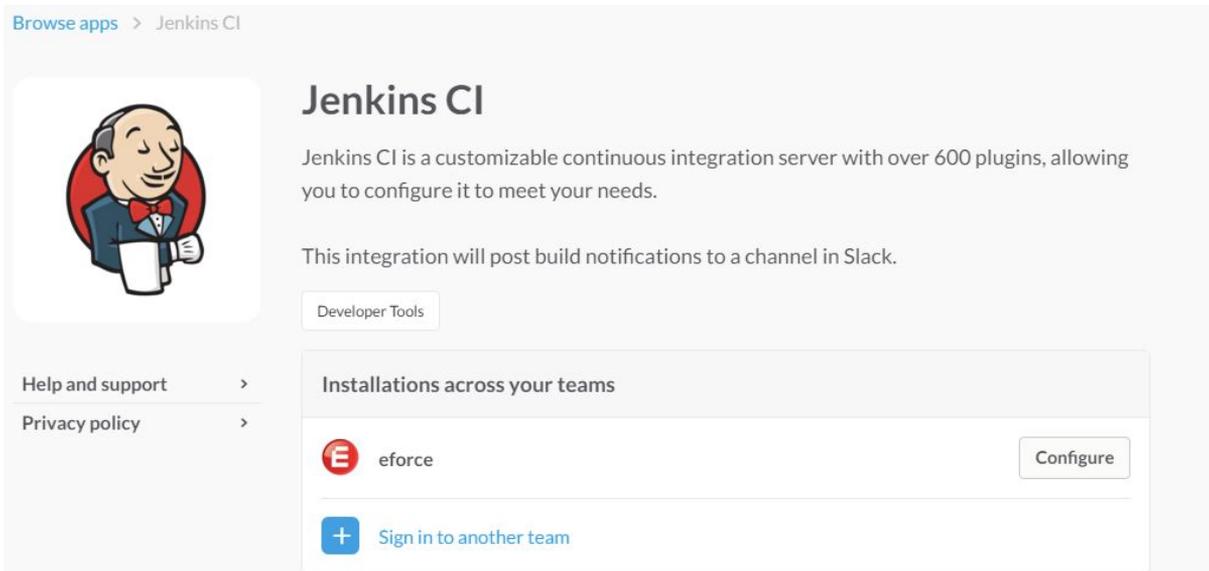


Figura 27 - Configurar Slack 3

Una vez seleccionado el equipo, seleccionamos "Add Configuration":



Figura 28 - Configurar Slack 4

Y después elegimos el canal en el que deseamos integrar Jenkins y seleccionamos "Add JEnkins CI integration":

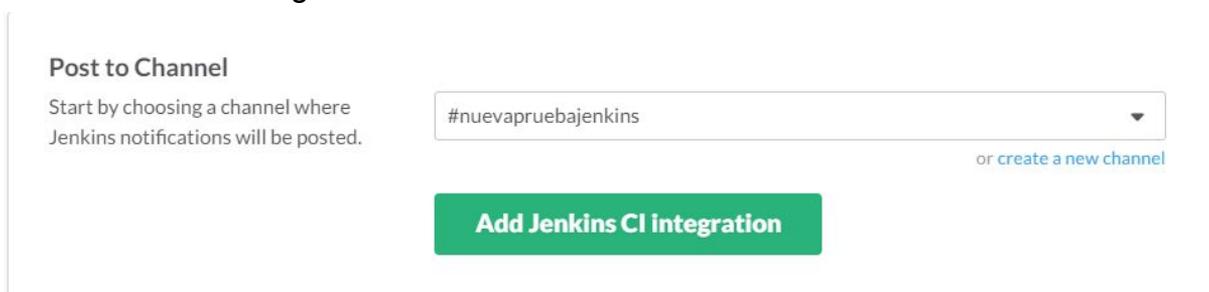


Figura 29 - Configurar Slack 5

En la siguiente pantalla es importante copiar el contenido de "Token":

Token

This token is used as the key to your Jenkins CI integration.

Otl9IWGNCAUIPFdvtpM6f0GY

[Regenerate](#)

Figura 30 - Configurar Slack 6

También podremos editar el nombre que tendrá el bot en el canal, su icono y otras cosas opcionales. Una vez acabados seleccionamos “Save Settings”.

En la parte de Jenkins:

En la sección “Manage Jenkins”, elegimos “Manage Plugins”:

The screenshot shows the Jenkins 'Manage Jenkins' interface. On the left sidebar, 'Manage Jenkins' is highlighted with a red box. The main content area is titled 'Manage Jenkins' and contains a warning message: 'Unsecured Jenkins allows anyone on the network to launch processes on your behalf. Consider at least enabling authentication to discourage misuse.' Below the warning are several configuration options: 'Configure System' (wrench icon), 'Configure Global Security' (lock icon), 'Reload Configuration from Disk' (refresh icon), 'Manage Plugins' (puzzle pieces icon, highlighted with a red box), and 'System Information' (monitor icon). The 'Manage Plugins' option includes the text: 'Add, remove, disable or enable plugins that can extend the functionality of Jenkins. (updates available)'. On the left sidebar, there are also sections for 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing '1 Idle' and '2 Idle').

Figura 31 - Enlazar Jenkins con Slack 1

Allí seleccionamos e instalamos (Install without restart) “Slack Notification Plugin”:

The screenshot shows the description for the 'Slack Notification Plugin'. The title is 'Slack Notification Plugin' and the description is 'This plugin is a Slack notifier that can publish build status to Slack channels.'

Figura 32 - Enlazar Jenkins con Slack 2

Una vez instalado el plugin, vamos a “Manage Jenkins” - “Configure System”. Allí, vamos al apartado “Global Slack Notifier Settings”, donde debemos insertar el subdominio del equipo, el token generado en el apartado anterior y el canal donde queremos recibir avisos. Seleccionamos “Apply” y después “Save”.

Global Slack Notifier Settings

| | | |
|-------------------|---|---|
| Team Subdomain | <input type="text" value="eforce-team"/> | ? |
| Integration Token | <input type="text" value="0t19IWGNCAUIPFdvtpM6f0GY"/> | ? |
| Channel | <input type="text" value="nuevapruerbajenkins"/> | ? |
| Build Server URL | <input type="text" value="/"/> | ? |

Figura 33 - Enlazar Jenkins con Slack 3

Después vamos a nuestro proyecto y entramos en la opción “Configure”. Allí, en el apartado “Post build action”, clicamos “Add post-build action” y seleccionamos “Slack Notifications”:

A screenshot of the Jenkins 'Add post-build action' dropdown menu. The menu is open, showing a list of actions. The 'Slack Notifications' option is highlighted with a blue border. Below the list is a button labeled 'Add post-build action' with a downward arrow. At the bottom of the screenshot are two buttons: 'Save' (dark blue) and 'Apply' (light grey).

- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Publish JUnit test result report
- Publish Javadoc
- Publish MSTest test result report
- Record Emma coverage report
- Record fingerprints of files to track usage
- Git Publisher
- E-mail Notification
- Slack Notifications**

▼

Figura 34 - Enlazar Jenkins con Slack 4

Allí podemos seleccionar en qué casos deseamos que Slack nos notifique:

Post-build Actions

- ☰ **Slack Notifications**
- Notify Build Start
 - Notify Aborted
 - Notify Failure
 - Notify Not Built
 - Notify Success
 - Notify Unstable
 - Notify Back To Normal



Figura 35 - Enlazar Jenkins con Slack

Seleccionamos “Apply” y después “Save”.
Ya deberíamos tener slack integrado y funcionando.



Figura 36 - Ejemplo de funcionamiento del bot de Slack

Anexo III (Crear un servicio FTP en Internet Information Services - IIS y enlazarlo a Jenkins):

Creación del servicio FTP:

Arrancamos Internet Information Service de Windows y en el servidor que tenemos (en nuestro caso es “EForce Server”) hacemos clic derecha y elegimos “Add Ftp site...”:

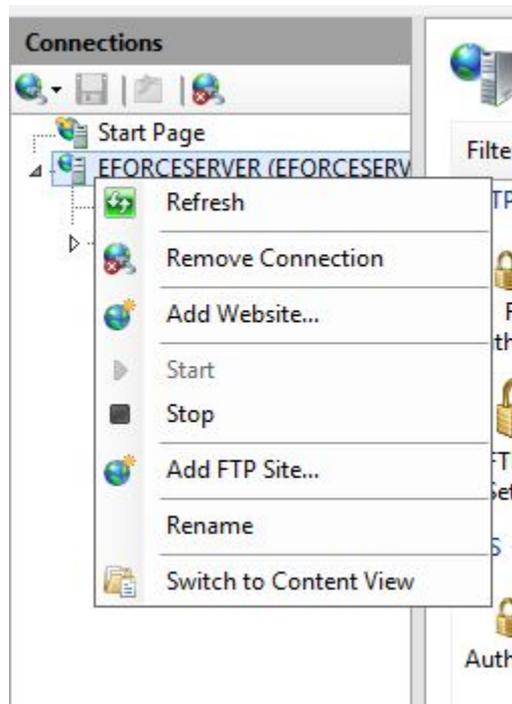


Figura 37 - Crear un sitio ftp 1

Se nos abrirá una ventana de diálogo en el que tenemos que introducir el dominio y la ruta física de la página ftp:

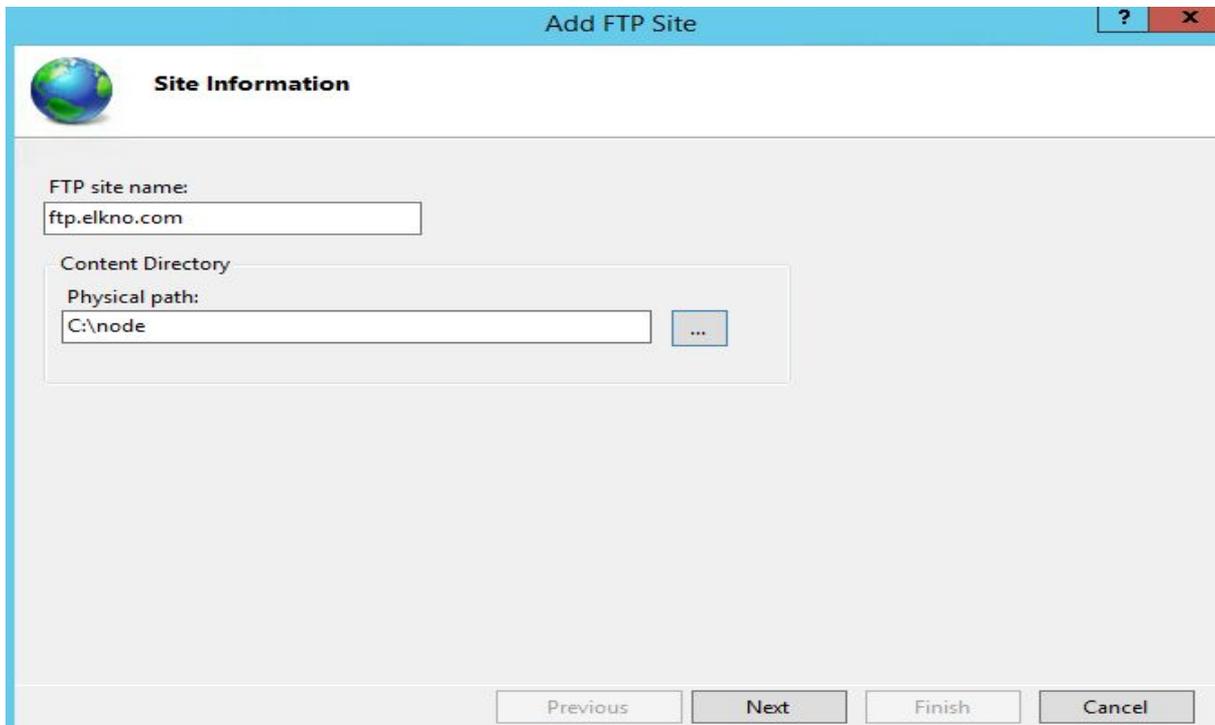


Figura 38 - Crear un sitio en ftp 2

Seleccionamos "Next" y nos pedirá que le asignemos un puerto y si queremos usar algún certificado de seguridad. En nuestro caso usaremos el puerto 21, que es el puerto por defecto para ftp y elegimos "No SSL" al no disponer de un certificado de seguridad. Seleccionamos "Next":

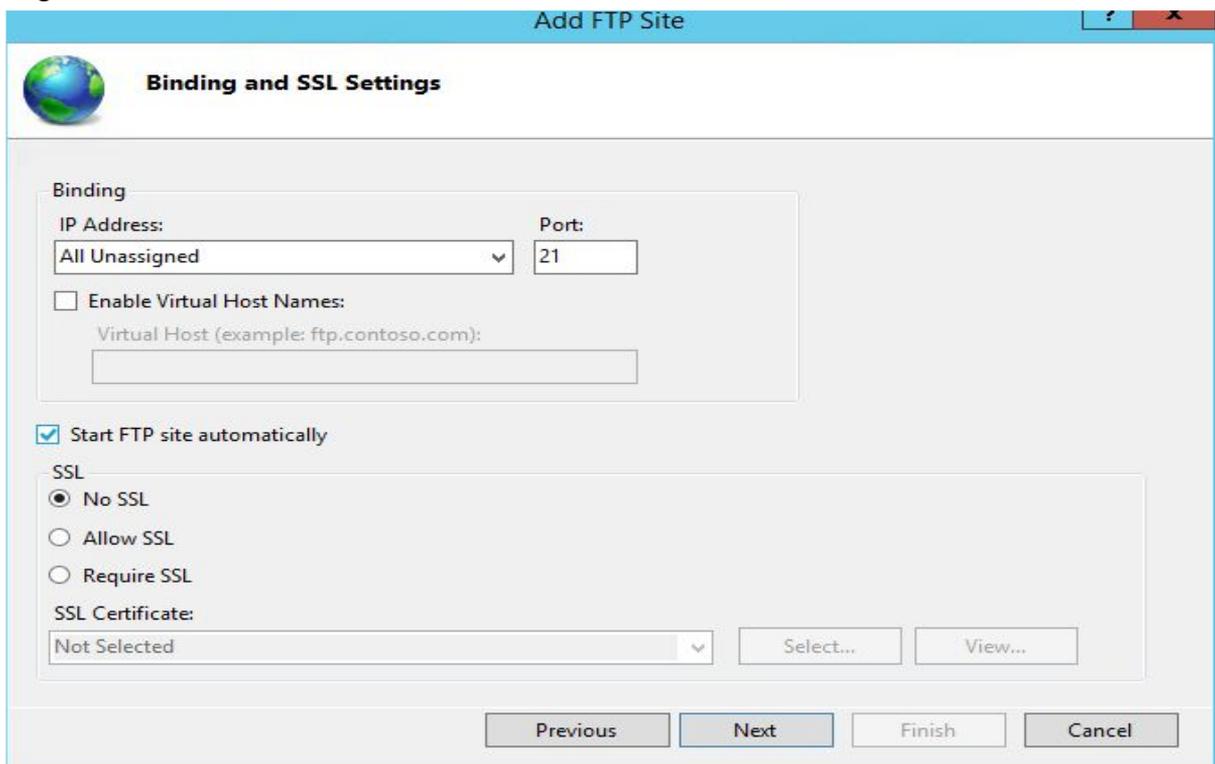


Figura 39 - Crear un sitio ftp 3

En el siguiente diálogo tenemos que configurar la autenticación del sitio. En nuestro caso seleccionamos la opción “Basic”, en “Allow access to:” seleccionamos “All users” y clicamos en los permisos “Read” y “Write” para dar acceso de lectura y escritura a todos los usuarios. Seleccionamos “Finish”:

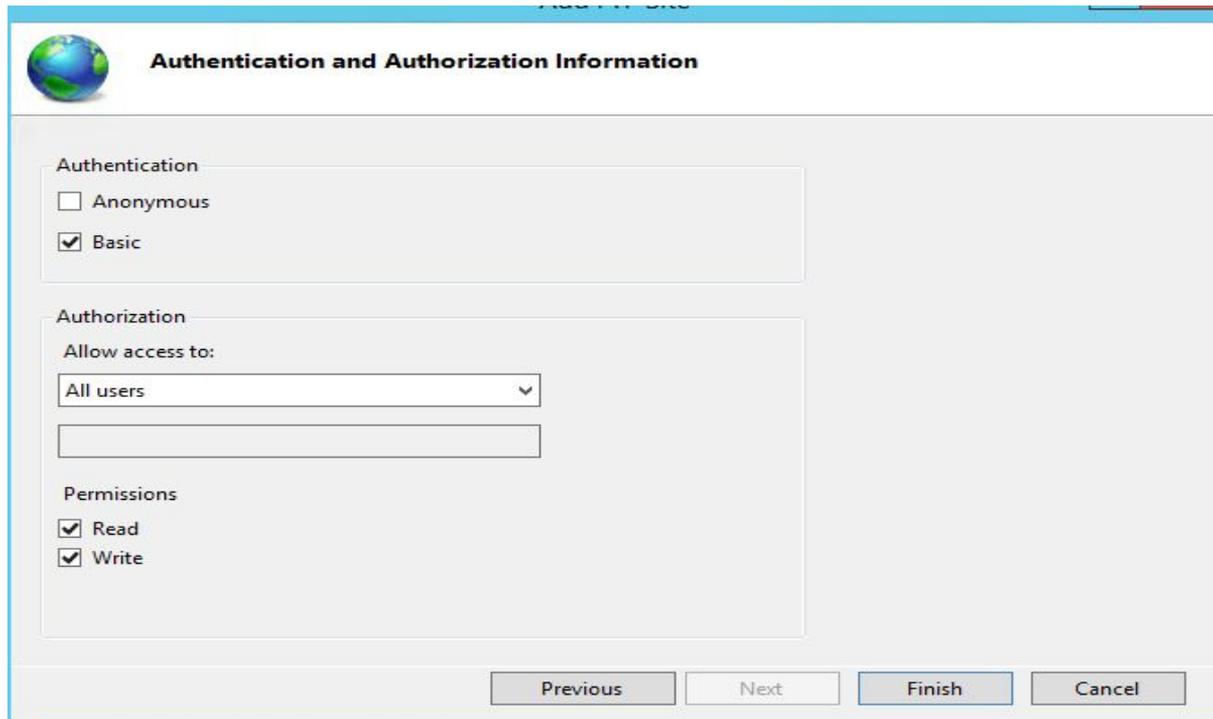


Figura 40 - Crear un sitio ftp 4

Una vez completados los pasos anteriores, vamos a “FTP Firewall Support” y dentro añadimos la IP Externa de la máquina donde se está ejecutando el Internet Information Service. En nuestro caso es “52.169.164.152”. Seleccionamos “Apply” en la parte derecha del menú:

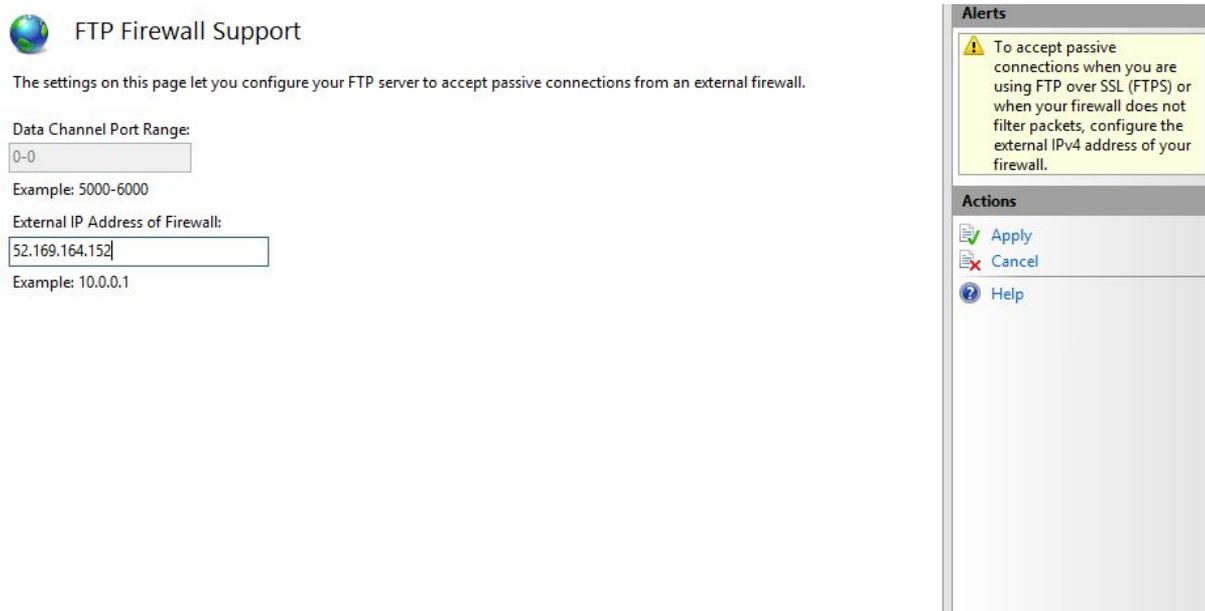


Figura 41 - Crear un sitio ftp 5

También hay que comprobar si el puerto 21 (o el haya sido asignado al sitio ftp) está abierto en el firewall de Windows. Para eso vamos a la configuración de firewall de Windows, y en “Inbound Rules” comprobamos que el puerto está activo. Sino, habrá que añadirlo con la opción “New Rule...”:

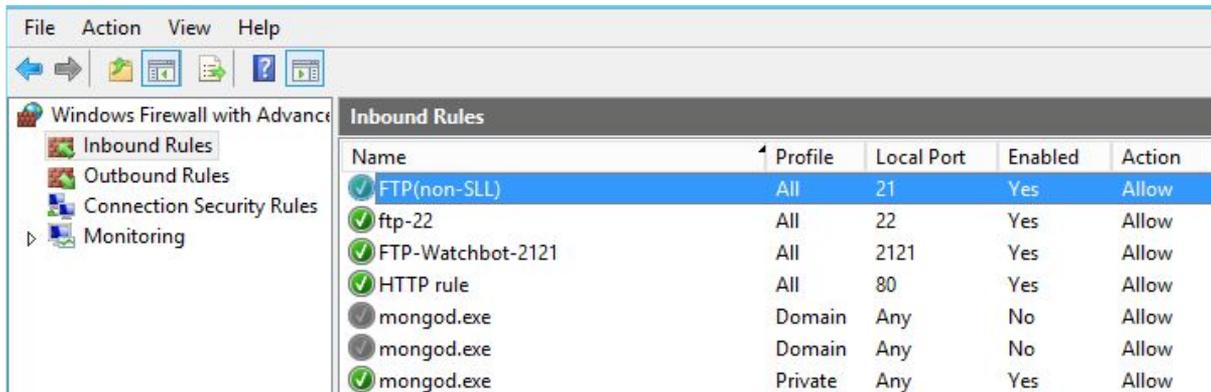


Figura 42 - Crear un sitio ftp 6

Por último, arrancamos el sitio FTP en la opción “Start” del menú de la derecha o seleccionamos “Restart” si ya estaba en ejecución:

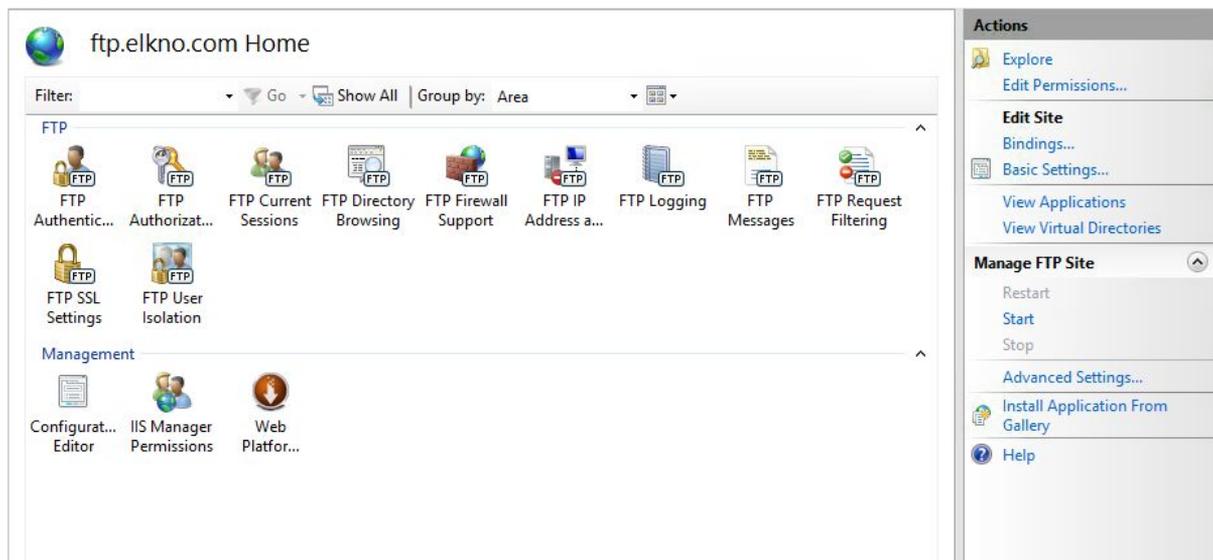


Figura 43 - Crear un sitio ftp 7

Enlazar Jenkins al sitio ftp:

Para enlazar Jenkins al sitio ftp que hemos creado, vamos a utilizar el plugin “FTP publisher plugin”. Para instalarlo vamos “Manage Jenkins” en el menú principal de Jenkins y seleccionamos “Manage Plugins”:



Figura 44 - Enlazar Jenkins al sitio ftp 1

Buscamos “ftp publisher”, seleccionamos “FTP publisher plugin” y seleccionamos “Install without restart”:



Figura 45 - Enlazar Jenkins al sitio ftp 2

Siguiente, entramos en “Manage Jenkins” -> “Configure System” y en el apartado “FTP repository hosts” seleccionamos Add y rellenamos los campos. En el campo “Profile Name” insertamos el nombre del perfil, en “hostname” añadimos la ruta al sitio ftp, en “Port” escribimos el puerto del sitio ftp (por defecto se usa el puerto 21). El campo “TimeOut” es opcional y se puede dejar a 0. En el campo “Root Repository Path” escribimos la ruta de nuestro proyecto en el sitio ftp (pueden coexistir varios proyectos en un mismo sitio ftp). En “User Name” y “Password” insertamos los credenciales de acceso al servicio ftp:



Figura 46 - Enlazar Jenkins al sitio ftp 3

Se pueden añadir más de un perfil en Jenkins.

Después vamos a nuestro proyecto, seleccionamos “Configure” y en el apartado “Post-build action” seleccionamos “Add post-build action” y en la lista resultante seleccionamos “Publish artifacts to FTP”:

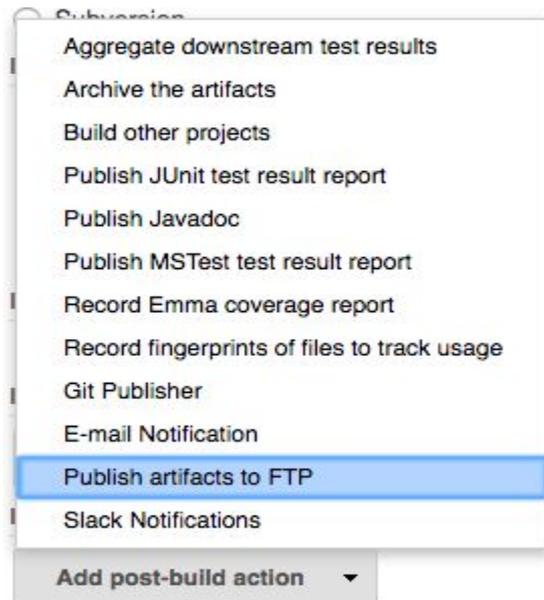


Figura 47 - Enlazar Jenkins al sitio ftp 4

En la lista “FTP site” seleccionamos el perfil deseado (y que hemos creado en “Configure System”).

En “Files to upload”, en el campo “Source” escribimos la ruta del fichero (o de los ficheros) que deseamos subir. La ruta es relativa a la carpeta “workspace” del proyecto dentro de Jenkins. En el campo “Destination” escribimos la ruta donde queremos guardar los ficheros dentro del sitio ftp. La ruta de destino empieza desde la ruta que hemos añadido en “Root repository path” en “Configure System”:

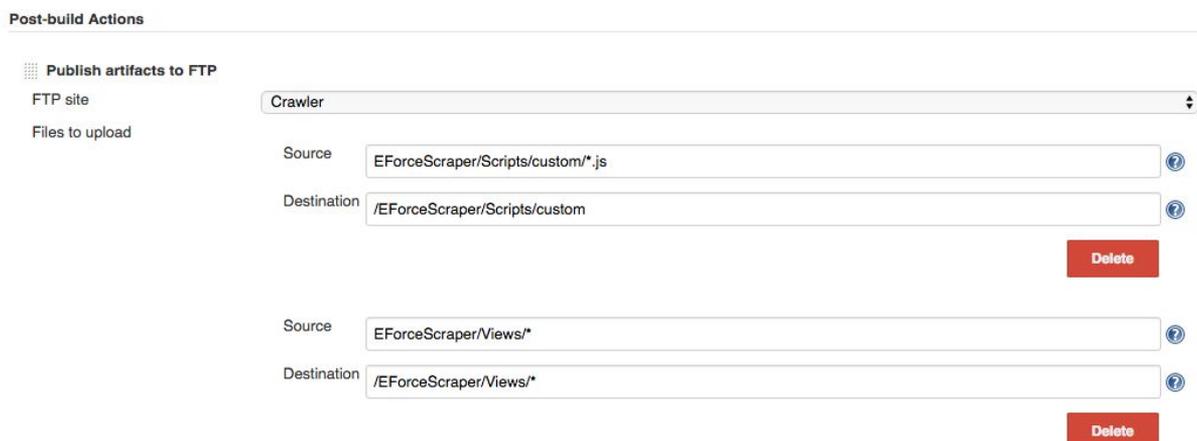
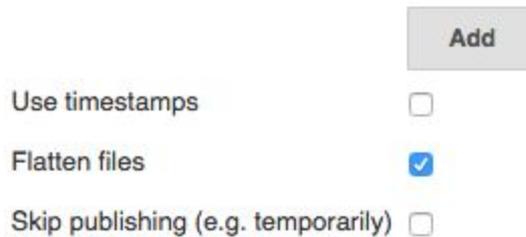


Figura 48 - Enlazar Jenkins al sitio ftp 5

También hay que activar la opción “Flatten files”, para que se copien sólo los ficheros, para que el plugin no copie los subdirectorios.



Use timestamps

Flatten files

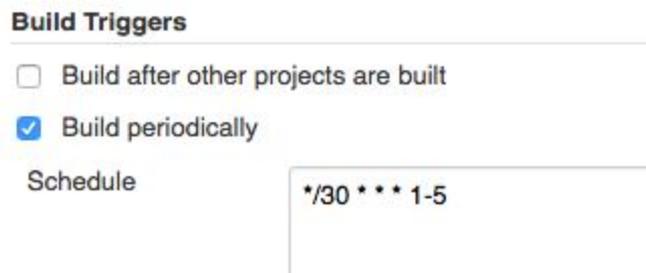
Skip publishing (e.g. temporarily)

Figura 49 - Enlazar Jenkins al sitio ftp 6

Después seleccionamos “Apply” y “Save”.

Programar Jenkins para que se ejecute periódicamente:

En el proyecto, seleccionamos “Configure”. En el apartado “Build Triggers”, seleccionamos la opción “Build periodically”. Se nos abrirá un cuadro y allí debemos insertar cuando queremos ejecutar el proyecto. El formato de sintaxis usado es similar a cron. Debe contener cinco campos, separados por espacio o tabulación. El primer campo el minuto (en qué minutos de una hora queremos ejecutarlo), el segundo campo es la hora del día, el tercer campo es el día del mes, el cuarto es el mes y el último campo es el día de la semana (1 es Lunes y 7 es el Domingo).



Build Triggers

Build after other projects are built

Build periodically

Schedule

Figura 50 - Ejemplo 1: Jenkins comprobará el proyecto cada 30 minutos, de lunes a viernes.

Bibliografia:

www.wikipedia.org

<https://jenkins.io/>

<http://www.iis.net/>

<https://htmlagilitypack.codeplex.com/>