



UNIVERSIDADE DA CORUÑA

Recuperación de Información y Web Semántica
Práctica de Recuperación de Información

José Manuel González Núñez
manuel.gonzalez1@udc.es

1. Tema y Motivación

Este documento aborda el trabajo llevado a cabo en relación a la práctica de *Recuperación de Información* de la asignatura de *Recuperación de Información y Web Semántica*.

Dicha práctica se ha centrado en implementar un buscador sobre las diferentes asignaturas que figuran en la guía docente[7] de la universidad de la Coruña. Este buscador posee una interfaz gráfica desde la cual se pueden lanzar consultas de búsqueda, así como filtrar asignaturas por diferentes aspectos.

Se ha elegido esta temática debido a la dificultad que existe cuando se desea buscar una asignatura dentro de la guía docente. El buscador implementado permitirá visualizar aquellas asignaturas que mantienen relación entre sí, además ofrecerá la posibilidad de hacer una búsqueda para hallar una asignatura concreta de manera inmediata.

La práctica se ha dividido en tres etapas claramente definidas:

1. Una primera etapa (**crawleado**) donde se obtiene la información de las páginas de cada una de las asignaturas.
2. La segunda etapa (**indexación**) consiste en indexar la información (obtenida en la etapa anterior) en un motor de búsqueda.
3. En la etapa final (**visualización**) se implementará la interfaz gráfica desde la cual se lanzarán las consultas. Esta interfaz gráfica se conectará con el motor de búsqueda para obtener la información sobre las asignaturas.

2. Tecnologías utilizadas

En este capítulo se describen las herramientas tecnológicas de cada una de las etapas descritas en el capítulo anterior.

Hay que destacar que el proyecto, en su totalidad, se ha desarrollado en un entorno Windows, por lo que la explicación de los diferentes aspectos de instalación de las herramientas y su uso, se explicará teniendo en cuenta este aspecto.

2.1. Crawleado

En la etapa de 'crawleado' se ha optado por utilizar el framework open source **Scrapy**[10] (versión 1.5.1). Este framework, escrito en Python[4] (versión 2.7.15), permite agilizar la extracción de datos de páginas web mediante una API o actuar como un rastreador web de propósito general.

Para la instalación de Scrapy, en Windows, se ha empleado la distribución open source **Anaconda**[6] (recomendada a través de la web de Scrapy). Esta distribución permite simplificar el despliegue y administración de paquetes de software, actuando como un sistema de gestión de paquetes. La versión que se ha utilizado en la realización de esta práctica ha sido la 5.2.2.

Este conjunto de herramientas y frameworks permite, de manera rápida, la instalación del crawler y su puesta en funcionamiento. Serán necesarios realizar los siguientes pasos:

1. Creación de un **nuevo entorno** en Anaconda. Ver *figura 1*.

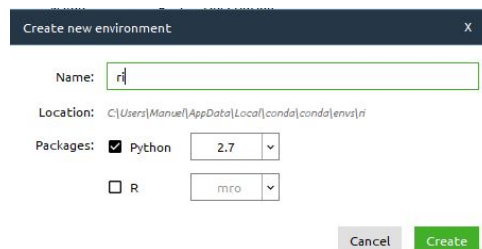


Figura 1: Creación de un nuevo entorno con Anaconda.

2. Desde la terminal del entorno creado, se instalará **openssl (v1.0.2)** con el comando

conda install openssl=1.0.2

3. Desde la terminal del entorno, se instalará **Scrapy** con el comando

conda install scrapy

Gracias a los anteriores pasos, Scrapy está instalado y listo para usarse. En la sección 4 se explica cómo se ha configurado Scrapy y cómo funciona para obtener la información de las diferentes asignaturas de la guía docente.

2.2. Indexación

Para la etapa de indexación de la información obtenida en la fase de crawling, se ha optado por el servidor de búsqueda **Elasticsearch**[1] (versión 6.4.2) basado en Lucene¹.

Elasticsearch ofrece un motor de búsqueda de texto completo y con una interfaz RESTful empleando objetos JSON. Esto es importante ya que la información extraída en la etapa del crawling estará en formato JSON, y la interfaz visual del buscador se comunicará con el motor de búsqueda de Elasticsearch a través de su interfaz RESTful.

La elección de esta herramienta se basa en que al estar desarrollada en Java, es compatible con todas las plataformas donde lo sea Java. Por otro lado, la velocidad de respuesta de su motor de búsqueda es muy elevada, además emplea objetos JSON, por lo que es fácil invocar desde otros lenguajes de programación.

En cuanto a la escalabilidad, Elasticsearch es capaz de dividir índices en fragmentos y cada uno de estos fragmentos podrá poseer réplicas.

La instalación de Elasticsearch en un entorno Windows es trivial. Simplemente basta con descargarse el instalador, ejecutarlo y mantener las opciones por defecto. Es necesario tener instalada una **versión de 64 bits de Java**.

Durante el desarrollo de la práctica también se ha empleado la herramienta Kibana[8]. Esta herramienta da forma a los datos indexados en Elasticsearch a través de una interfaz de usuario visual, mediante la que se pueden visualizar y administrar.

Para ejecutar ElasticSearch bastará con lanzar el ejecutable “**elasticsearch.exe**” que se encuentra en la carpeta ‘*bin*’ donde se ha instalado Elastic. De la misma forma, para lanzar Kibana, bastará con ejecutar el archivo “**kibana.bat**” que se encuentra en la carpeta donde se ha instalado.

2.3. Visualización

En la última etapa de la realización de esta práctica, se ha implementado una interfaz web que permita realizar búsquedas sobre la información de las asignaturas, así como filtrar dichas asignaturas en relación a determinados parámetros. Para ello se ha empleado la librería **ReactiveSearch**[9] que proporciona componentes de la interfaz de usuario de Elasticsearch para **React.js**[5].

La librería incorpora más de 25 componentes, *ver figura* :

- Listas.
- Rangos.
- Input para la búsqueda de un usuario.
- Pantallas de resultados.
- Etc.

El punto fuerte de esta librería, y de React.js, es que los componentes se pueden combinar para realizar búsquedas más complejas.

¹API de código abierto para recuperación de información.

Por otro lado, React.js, es una biblioteca de Javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo.

En primer lugar, para la utilización de estas librerías, será necesario instalar Node.js[2] (versión 8.12) y npm[3](versión 6.4.1). Para ello bastará con ejecutar el instalador de Node.js (npm también será instalado). Npm es un manejador de paquetes para Node.js, por lo tanto esta herramienta se empleará para instalar reactiveSearch:

```
npm install @appbaseio/reactivesearch
```

3. Información recuperada

La estructura de la información de cada una de las asignaturas de la guía docente se divide en los siguientes apartados:

- **Nombre de la asignatura.**
- **Nombre de la titulación.**
- **Código de la asignatura.**
- Ciclo.
- **Cuatrimestre.**
- **Curso.**
- **Tipo.**
- **Créditos.**
- Idioma.
- Prerrequisitos.
- **Departamento.**
- **Coordinación (nombres - emails).**
- **Profesorado (nombres - emails).**
- Web.
- **Descripción.**
- **Competencias del título (código - competencia).**
- Resultados del aprendizaje.
- **Contenidos (temas - subtemas).**
- Planificación.
- Metodologías.
- Atención personalizada.
- Evaluación.
- Fuentes de información.
- Recomendaciones.

Siendo los apartados subrayados los que se han tenido en cuenta a la hora de construir el buscador. En la *figura 2* se pueden ver dichos apartados.




La obtención de la información de los apartados restantes se llevaría a cabo siguiendo el mismo procedimiento (descrito a continuación).

Mestrado Universitario en Enxeñaría Informática (plan 2012)

Asignaturas

Recuperación da información e web semántica

gallego castellano english

Datos Identificativos		2018/19
Asignatura (*)	Recuperación da información e web semántica	Código 614502010
Titulación Mestrado Universitario en Enxeñaría Informática (plan 2012)		
Descriptor	Ciclo Mestrado Oficial	Período 1º cuatrimestre
	Curso Primeiro	Tipo Obligatoria
		Créditos 6
Idioma	Castelán	
Prerrequisitos		
Departamento	Computación	
Coordinación	Barreiro Garcia, Álvaro	Correo electrónico alvaro.barreiro@udc.es
Profesorado	Barreiro Garcia, Álvaro Fernández Iglesias, Diego Parapar López, Javier Vázquez Naya, José Manuel	Correo electrónico alvaro.barreiro@udc.es diego.fernandez@udc.es javier.parapar@udc.es jose.manuel.vazquez.naya@udc.es
Web		
Descrición xeral	Os modelos, técnicas e algoritmos de recuperación de información estudados nesta materia permitirán aos estudantes comprender a arquitectura dos Search Engines para a web. Ademais os contidos prácticos da mesma capacitaránlles para construír os seus propios buscadores para traballar sobre repositorios de documento ou a web. Ademais durante os últimos anos houbo un interese crecente en idear unha web semántica a partir de meta-datos e anotacións. Unha web baseada en documentos xml e tags, meta-datos e esquemas, sen dúbida facilitaríala os enormes retos aos que se enfronta a recuperación de información web. Nesta materia abórdanse tamén os modelos, técnicas e algoritmos de maior impacto desenvolvidos nos últimos anos co obxectivo de materializar unha web semántica. A Recuperación de Información en grandes coleccións de documentos e na web expón enormes retos (volumen de datos, datos distribuídos, alta porcentaxe de datos volátiles, datos non estruturados e redundantes, heteroxeneidade, calidade dos datos e confianza) e a Web Semántica parte xa do gran reto da extracción de información cando os meta-datos non son expostos publicamente e expón novos retos como os do matching de ontoloxías, resolución de entidades ou unha dificultade maior en canto á heteroxeneidade e calidade dos datos e á indexación e procura semántica. Por todo iso a Recuperación de Información e a Web semántica constitúen un dos campos de mellores saídas profesionais en informática con oportunidades de negocio e emprego non só nas grandes compañías de Search Engines senón tamén en moitas pequenas e medianas compañías.	
Competencias do título	Resultados de aprendizaxe	Contidos
Planificación	Metodoloxías	Atención personalizada
Avaliación	Fontes de información	Recomendacións

(*)A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente de acordo coa normativa vixente que establece o proceso de elaboración de guías

Figura 2: Estructura de la información de las asignaturas.

4. Crawleado de la información

Como se ha dicho en la **sección 2.1**, el proceso de obtención de información se ha realizado con Scrapy. Una vez instalada esta herramienta, se procederá a crear un nuevo proyecto con el comando

scrapy startproject <nombreProyecto>

Dicho comando creará la estructura de directorios y ficheros necesarios para un proyecto Scrapy.

```

carpetaProyecto/
├─ scrapy.cfg
├─ nombreProyecto/
│   └─ __init__.py
│       └─ items.py
│           └─ middlewares.py
│               └─ pipelines.py
│                   └─ settings.py
│                       └─ spiders/
│                           └─ __init__.py

```

El funcionamiento de esta herramienta se basa en la creación de **'spiders'**, los cuales actúan como crawlers autocontenidos que realizarán una serie de acciones sobre un determinado contexto.

En este paso se creará el spider **'guiaDocente.py'** (*adjunto a la entrega de este proyecto*) dentro de la carpeta 'spiders'. Dicho spider ha de ser una subclase de **CrawlSpider**.

El crawler irá 'navegando' por los diferentes enlaces de la página <https://www.udc.es/ensino/guiasdocentes/>, que actuará como página de 'semilla'. El proceso de navegación entre las páginas será el siguiente:

1. En primer lugar entrará en las páginas de **cada uno de los centros y escuelas** de la universidad.
2. Seguidamente se accederá a las **titulaciones** de cada centro.
3. En tercer lugar se accederá al **listado de las asignaturas** de cada una de las titulaciones.
4. Y por último, el crawler entrará a la **página web de cada asignatura**, inspeccionando también los subapartados de *'competencias'* y *'contenidos'*.

Para realizar esta navegación a través de las diferentes páginas de la guía docente, es necesaria la definición de reglas. Dichas reglas irán comprobando los enlaces, y en función del tipo de enlace se llamará a una función determinada para extraer la información que contiene dicho enlace. Por ejemplo, un enlace que cumpla la expresión regular

r'.*&asignatura=[a-zA-Z0-9]{9}&any_academic=[0-9]{4}_[0-9]{2}\$'

se corresponderá con la página web de una asignatura concreta, como por ejemplo

...ensenyament=631G01&asignatura=631G01102&any_academic=2018_19

Cuando el enlace se corresponda al de una página que almacena la información de una asignatura, entonces se llamará a la función **'parse_asignatura'**. Esta función creará una item llamado **AsignaturaItem** al cual se le irá añadiendo la información de los diferentes campos extraídos (ver *sección 3*). Es necesario definir en el fichero **'items.py'** la estructura del item.

Además, se accederá a la información de los subapartados 'contenidos' y 'competencias' de las páginas de cada asignatura para incorporar dicha información al item creado.

Finalmente, una vez extraída toda la información de la asignatura, se pasará el item a un **'pipeline'** para procesar su volcado en un fichero en formato JSON.

Se ha definido un pipeline, llamado **'JsonPipeline'** en el fichero **'pipelines.py'**. El pipeline creará una carpeta **'results'** en el directorio del proyecto. Dentro de dicha carpeta se crearán subcarpetas para cada uno de los centros de la universidad y dentro de cada una de ellas se almacenarán los archivos JSON con información de las asignaturas para cada titulación.

```
carpetaProyecto/  
├─ scrapy.cfg  
├─ crawlGuiaDocente/  
└─ results/  
    ├─ Escola Politècnica Superior/  
    │   ├─ Mestrado Universitario en Enxeñaría Naval e Oceánica (plan 2018)  
    │   ├─ Grao en Enxeñaría Mecánica  
    │   └─ ...  
    ├─ Escola Técnica Superior de Arquitectura/  
    └─ ...
```

Además, los ficheros JSON seguirán un formato concreto para indexar la información de cada asignatura en ElasticSearch. Para poder indexar un fichero JSON que contiene un conjunto de información de diferentes asignaturas, es necesario que cada objeto JSON de una asignatura, sea precedido por un identificador de documento (en este caso se utilizará el código de las asignaturas).

El formato de los ficheros JSON que almacenan la información de las diferentes asignaturas para cada una de las titulaciones se puede ver a continuación.

Listing 1: Formato de los ficheros JSON.

```
...  
{ "index": { "_id": "730496024" } }  
{ "codigo": "730496024", "competencias" : ..... }  
{ "index": { "_id": "730496011" } }  
{ "codigo": "730496011", "competencias" : ..... }  
{ "index": { "_id": "730496013" } }  
...
```


Para habilitar dicho pipeline, será necesario configurar la cláusula **ITEM_PIPELINES** en el fichero **'settings.py'**.

```
ITEM_PIPELINES = { 'crawlGuiaDocente.pipelines.JsonPipeline': 300, }
```

En este punto del desarrollo del proyecto, la programación del crawler y la configuración de Scrapy ya estaría finalizada, por lo que es posible lanzar el spider **'guia-Docente'** para que se capture la información.

La carpeta adjunta a la entrega de esta memoria contiene toda la configuración necesaria, por lo que sólo será necesario ejecutar el siguiente comando en el entorno donde se ha instalado Scrapy y openssl para empezar a crawl.

scrapy crawl guiaDocente - -loglevel INFO

En la **figura 3** podemos ver los diferentes mensajes durante el transcurso de la ejecución del crawler.

```
C:\Windows\system32\cmd.exe - scrapy crawl guiaDocente - -loglevel INFO
2018-11-14 19:46:53 [root] INFO: [2.125] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Sistemas de Control (730496227)
2018-11-14 19:46:53 [root] INFO: [2.15X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Sistemas de Propulsión (730496218)
2018-11-14 19:46:53 [root] INFO: [2.17X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Sistemas de Propulsión (730496016)
2018-11-14 19:46:53 [root] INFO: [2.20X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Novas Tecnoloxías de Enxeñaría Naval (730496224)
2018-11-14 19:46:53 [root] INFO: [2.23X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Proceso Integral do Proxecto do Buque (730496281)
2018-11-14 19:46:53 [root] INFO: [2.25X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Oceanografía (730496208)
2018-11-14 19:46:53 [root] INFO: [2.28X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Métodos numéricos aplicados a medios continuos (730496022)
2018-11-14 19:46:53 [root] INFO: [2.31X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Máquinas e Motores Térmicos Mariños (730496219)
2018-11-14 19:46:54 [root] INFO: [2.33X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Máquinas e Motores Térmicos marinos (730496017)
2018-11-14 19:46:54 [root] INFO: [2.36X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Proceso Integral de construción de buques (730496006)
2018-11-14 19:46:55 [root] INFO: [2.39X] --> Titulación: Grupos en Socioloxía Asignatura: Antropoloxía social e cultural (615601102) Asignatura: Estructuras navais (730496021)
2018-11-14 19:46:55 [root] INFO: [2.41X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Dinámica do buque (730496004)
2018-11-14 19:46:55 [root] INFO: [2.44X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Equipos e Servizos (730496220)
2018-11-14 19:46:56 [root] INFO: [2.46X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Ampliación de matemáticas (730496015)
2018-11-14 19:46:56 [root] INFO: [2.49X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Contabilidade, Planificación e Control de Custos (730496225)
2018-11-14 19:46:56 [root] INFO: [2.52X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Estructuras Navais (730496223)
2018-11-14 19:46:56 [root] INFO: [2.54X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Ampliación de Hidrostática e Hidrodinámica (730496222)
2018-11-14 19:46:56 [root] INFO: [2.57X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Ampliación de projecto de buques (730496001)
2018-11-14 19:46:56 [root] INFO: [2.60X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Climatización e Refrixeración (730496226)
2018-11-14 19:46:56 [root] INFO: [2.62X] --> Titulación: Enxeñeiro Industrial Asignatura: Tecnoloxía Eléctrica (730211500)
2018-11-14 19:46:56 [root] INFO: [2.65X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Ampliación de hidrostática e hidrodinámica (730496020)
2018-11-14 19:46:56 [root] INFO: [2.68X] --> Titulación: Enxeñeiro Industrial Asignatura: Sistemas de Prefabricación (730211510)
2018-11-14 19:46:56 [root] INFO: [2.70X] --> Titulación: Enxeñeiro Industrial Asignatura: Mecánica da Fractura (730211510)
2018-11-14 19:46:57 [root] INFO: [2.73X] --> Titulación: Enxeñeiro Industrial Asignatura: Topografía e Fotogrametría (730211512)
2018-11-14 19:46:57 [root] INFO: [2.76X] --> Titulación: Mestrado Universitario en Enxeñaría Naval e Oceanica (plan 2018) Asignatura: Ampliación de hidrostática e hidrodinámica (730496020)
2018-11-14 19:46:57 [root] INFO: [2.78X] --> Titulación: Enxeñeiro Industrial Asignatura: Planificación Enerxética (730211515)
2018-11-14 19:46:57 [root] INFO: [2.81X] --> Titulación: Enxeñeiro Industrial Asignatura: Construción e Arquitectura Industrial II (730211513)
2018-11-14 19:46:57 [root] INFO: [2.84X] --> Titulación: Enxeñeiro Industrial Asignatura: Tecnoloxía Nuclear (730211516)
2018-11-14 19:46:57 [root] INFO: [2.86X] --> Titulación: Enxeñeiro Industrial Asignatura: Administración de Empresas (730211509)
2018-11-14 19:46:57 [root] INFO: [2.89X] --> Titulación: Enxeñeiro Industrial Asignatura: Deseño Asistido por Ordenador (730211505)
2018-11-14 19:46:57 [root] INFO: [2.92X] --> Titulación: Enxeñeiro Industrial Asignatura: Tecnoloxía Frigorífica (730211500)
2018-11-14 19:46:57 [root] INFO: [2.94X] --> Titulación: Enxeñeiro Industrial Asignatura: Regulación Automática (730211504)
2018-11-14 19:46:57 [root] INFO: [2.97X] --> Titulación: Enxeñeiro Industrial Asignatura: Organización da Produción (730211507)
```

Figura 3: Ejecución del crawler.

El proceso de crawling completo tarda alrededor de 14 minutos, por lo que se ha añadido a la entrega la carpeta **"results/"** que contiene toda la información recuperada (resultado del proceso completo de crawling).

5. Indexación

Una vez obtenida la información de todas las asignaturas de la guía docente, se ha procedido a su indexación en Elasticsearch. Se ha utilizado la herramienta Kibana en esta fase como soporte para la creación del índice y la comprobación de que la información ha sido indexada correctamente.

Tanto para la creación del índice como para la indexación se emplean peticiones HTTP (PUT o POST) sobre Elasticsearch. El contenido de dichas peticiones serán los ficheros de cada una de las titulaciones que contienen la información de sus respectivas asignaturas.

Antes de indexar la información, crearemos el índice indicando la estructura de los documentos que se indexarán. Esta estructura se corresponderá con la estructura de la información recabada de cada asignatura. En la definición de la estructura también se especificarán los **analizadores** que se emplearán para cada ‘campo’ de la asignatura.

A continuación podemos ver un fragmento de la definición de la estructura que seguirá el índice.

Listing 2: Estructura del índice.

```
{ "mappings": {  
  "_doc": {  
    "properties": {  
      "nombre_asignatura": {  
        "type": "text",  
        "analyzer" : "galician"  
      },  
      "nombre_titulacion": {  
        "type": "text",  
        "analyzer" : "galician",  
        "fields": {  
          "keyword": {  
            "type": "keyword",  
            "ignore_above": 256  
          }  
        }  
      },  
      "creditos": {  
        "type": "float"  
      },  
      "codigo": {  
        "type": "keyword"  
      },  
      ....  
      ....  
    }  
  }  
}
```

En el fragmento anterior podemos observar como el campo “**nombre_asignatura**” ha sido definido como tipo ‘**text**’. Este tipo de *fields* se pasan a través de un analizador para convertir la cadena en una lista de términos individuales antes de ser indexados. Este proceso de análisis permite a Elastic buscar palabras individuales dentro de cada field de texto completo.

El *field* “**nombre_titulacion**” también es del tipo ‘**text**’, pero posee el parámetro “field” en su interior. Este parámetro permite que el mismo valor de cadena se indexe de múltiples maneras para diferentes propósitos, como un campo para búsqueda y un campo múltiple para la clasificación y agregaciones.

El tipo ‘**keyword**’ se emplea para campos como direcciones de correo electrónico, nombres de host, códigos, etiquetas. Y por lo general se utilizan para el filtrado (encontrar todas las asignaturas de una titulación), para la clasificación y para las agregaciones. Los campos de tipo ‘keyword’ solo se pueden buscar por su valor exacto. Como se verá en la *sección 6*, se implementará un filtro por el campo “nombre_titulacion”, por lo que el **field** ‘**keyword**’ es necesario.

En cuanto el analizador para los *fields* de tipo ‘text’ se ha elegido el **analizador de gallego** (galician), puesto que la información obtenida está (o debería) escrita en gallego.

Este analizador, está incorporado en Elasticsearch y será el encargado de realizar el análisis sobre los *fields* de tipo ‘text’. El analizador dividirá el texto en ‘tokens’, luego convertirá todos los caracteres a minúsculas y por último eliminará las palabras frecuentes (por ejemplo, ‘de’, ‘un’, etc). Los términos resultantes después de aplicar el analizador serán indexados en el índice invertido.

A continuación se puede ver como actúa el analizador de gallego sobre la frase

Non busques nin boi de sábado nin home de luns

Los tokens obtenidos después de aplicar el analizador a este refrán son:

[busc, bo, sab, hom, lun]

El archivo “**mappingFieldsAsignatura.json**”, adjunto a la entrega de este proyecto, posee la definición completa de la estructura para el índice.

Para crear un índice con dicha estructura, bastará con hacer una petición **PUT <nombreIndice>** a la dirección de elasticsearch (si se está ejecutando en local, la dirección es *localhost:9200*). En el cuerpo de la petición se enviará el JSON que contiene el fichero “mappingFieldsAsignatura.json”.

Adjunto a la entrega de este proyecto, también se encuentra un pequeño programa, “**indexarAsignaturas.py**”², escrito en Python para crear el índice sin necesidad de realizar las peticiones de manera individual.

Este programa posee tres opciones:

- **python indexarAsignaturas.py -c <nombreIndice>** . Creará un índice con la estructura definida en “mappingFieldsAsignatura.json”
- **python indexarAsignaturas.py -i <nombreIndice>** . Indexará en el índice (previamente creado) todos los archivos .json que están en la carpeta ./results, creada tras la ejecución del crawler.

²Necesario tener el paquete de Python ‘requests’ (pip install requests).

- **python indexarAsignaturas.py -d <nombreIndice>** . Borra el índice.
- **python indexarAsignaturas.py -e <nombreIndice>** . Información del índice.

Una vez creado el índice, y con la ayuda del programa “indexarAsignaturas.py” se indexarán todos los documentos originados tras la ejecución del crawler. En la *figura 4* se puede observar el proceso de creación e indexación de la información gracias a la ayuda de este programa.

```
(prueba) C:\Users\Manuel\prueba>python indexarAsignaturas.py -c asignatura
Indice "asignatura" creado

(prueba) C:\Users\Manuel\prueba>python indexarAsignaturas.py -i asignatura
Indexados los documentos del fichero "Enxeñeiro Industrial.json"
Indexados los documentos del fichero "Enxeñeiro Naval e Oceánico.json"
Indexados los documentos del fichero "Grao en Arquitectura Naval.json"
Indexados los documentos del fichero "Grao en Enxeñaría en Propulsión e Servizos do Buque.json"
Indexados los documentos del fichero "Grao en enxeñaría en Tecnoloxías Industriais.json"
Indexados los documentos del fichero "Grao en Enxeñaría Mecánica.json"
Indexados los documentos del fichero "Grao en Enxeñaría Naval e Oceánica.json"
Indexados los documentos del fichero "Mestrado Universitario en Enxeñaría Industrial (plan 2018).json"
Indexados los documentos del fichero "Mestrado Universitario en Enxeñaría Naval e Oceánica (plan 2018).json"
Indexados los documentos del fichero "Mestrado Universitario en Materiais Complexos Análise Térmica"
Indexados los documentos del fichero "Máster Universitario en Deseño, Desenvolvemento e Comercialización"
Indexados los documentos del fichero "Grao en Arquitectura.json"
Indexados los documentos del fichero "Grao en Estudos de Arquitectura.json"
Indexados los documentos del fichero "Grao en Paisaxe.json"
Indexados los documentos del fichero "Mestrado Universitario en Arquitectura da Paisaxe Juana de Vega"
Indexados los documentos del fichero "Mestrado Universitario en Arquitectura.json"
Indexados los documentos del fichero "Mestrado Universitario en Rehabilitación Arquitectónica (Plan 2018).json"
Indexados los documentos del fichero "Enxeñeiro de Camiños, Canais e Portos.json"
Indexados los documentos del fichero "Grao en Enxeñaría de Obras Públicas.json"
Indexados los documentos del fichero "Grao en Tecnoloxía da Enxeñaría Civil.json"
```

Figura 4: Creación del índice e indexación de información.

Podemos ver la información del índice después de haber realizado la indexación, ver *figura 5*.

```
(prueba) C:\Users\Manuel\prueba>python indexarAsignaturas.py -e asignatura
health status index      uuid                                pri rep docs.count docs.deleted store.size pri.store.size
yellow open   asignatura 36mJlp11500bjDPYId12wg      5   1    3773           621      18.7mb      18.7mb
```

Figura 5: Información del índice tras la indexación.

El índice posee 3773 documentos, que se corresponden con las **3773 asignaturas** (*a día 19 de Noviembre de 2018*) impartidas en la Universidad de la Coruña.

6. Visualización

En la *sección 2.3* se ha hablado de las herramientas que se emplearán para aportar un buscador con interfaz web al índice creado.

En primer lugar será necesario configurar ElasticSearch para que ReactiveSearch se pueda conectar a el y obtener los resultados de las búsquedas. Por lo tanto, habrá que añadir las siguientes cláusulas *3* al fichero

C:\ProgramData\Elastic\Elasticsearch\config\elasticsearch.yml

Listing 3: Permitir conexiones desde React

```
http.cors.enabled: true
http.cors.allow-credentials: true
http.cors.allow-origin: "http://localhost:3000"
http.cors.allow-headers: X-Requested-With, X-Auth-Token, Content-Type,
Content-Length, Authorization, Access-Control-Allow-Headers, Accept
```

El proyecto Node creado para la interfaz web está contenido en la carpeta **“asignaturasearch”**. Situados dentro de dicha carpeta, se procederá a la instalación de reactiveSearch con el comando

npm install @appbaseio/reactivesearch

Una vez instalada dicha dependencia se podrá ejecutar el proyecto con el comando

npm start

Cuando se lance el proyecto de React, se abrirá una interfaz con el buscador, ver *figura 6*. **Se podrá acceder a la página de la guía docente de cada asignatura pulsando sobre su nombre.**

6.1. Funcionalidades incorporadas

A continuación se emplean los diferentes componentes utilizados en la interfaz web, y como estos, mejoran la forma de realizar consultas.

6.1.1. Buscador de texto

El buscador de texto principal, ver *figura 7*, provee al usuario de una entrada de texto para realizar una búsqueda que engloba todos los campos de las asignaturas.

Este componente se llama **“DataSearch”** en la librería ReactiveSearch y pPosee la opción de establecer pesos a los campos por lo que se busca, para que así ElasticSearch pueda calcular el score en relación a dichos pesos.

La relación peso-campo establecida es la siguiente:

- Nombre asignatura: 5
- Nombre titulación: 3

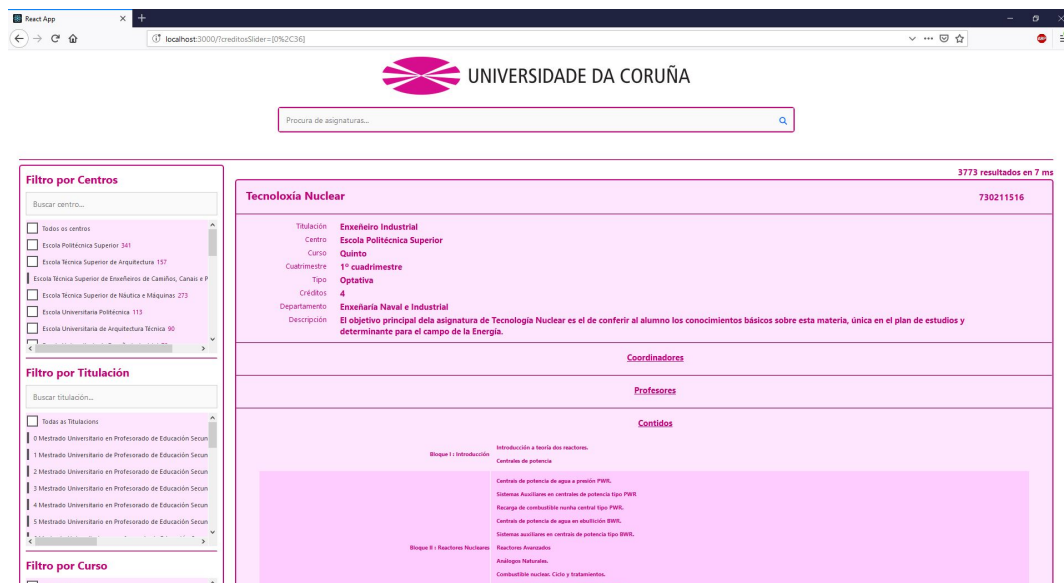


Figura 6: Visión global de la interfaz.



Figura 7: Entrada de texto.

- Nombre centro: 3
- Código: 4
- Cuatrimestre: 2
- Tipo: 2
- Curso: 2
- Nombre Coordinadores: 2
- Email Coordinadores: 2
- Nombre Profesores: 2
- Email Profesores: 2
- Departamento: 2
- Descripción: 1
- Código competencia: 1
- Descripción competencia: 1
- Subtema Contenido: 1
- Tema Contenido: 1

El resultado de la realización de una búsqueda siempre se mostrará en el orden del score que tiene cada resultado.

6.1.2. Barra de filtros y búsquedas realizadas

Se ha añadido el componente **“SelectedFilters”** para que el usuario sepa en todo momento que filtros tiene activos y si desea eliminar alguno de ellos. Ver *figura 8*.

búsqueda: indexación	X	Centro: Facultade de Informática	X	Titulación: Mestrado Universitario en Enx...	X	Borrar filtros
						1 resultados en 5 ms
Recuperación da información e web semántica						614502010
Titulación	Mestrado Universitario en Enxeñaría Informática (plan 2012)					
Centro	Facultade de Informática					
Curso	Primeiro					
Asignatura	401010101					

Figura 8: Filtros activos.

6.1.3. Filtros

En la interfaz se han añadido tres componentes **“MultiList”**. Estos componentes permiten la selección múltiple de los distintos valores de un campo. Es decir, actúan como filtros sobre un campo de la información de cada asignatura. Además reaccionan si otros filtros o entradas de búsqueda se modifican.

Los campos sobre los que se han añadido los filtros son:

- Nombre del centro.
- Nombre de la titulación.
- Curso de la asignatura.

En la *figura 9* se pueden ver dichos filtros.

Filtro por Centros

☐ Facultade de Ciencias da Educación 524
 ☐ Facultade de Ciencias da Saúde 95
 ☐ Facultade de Ciencias do Deporte e a Educación Física 67
 ☐ Facultade de Ciencias do Traballo 90
 ☐ Facultade de Dereito 94
 ☐ Facultade de Economía e Empresa 190
 ☐ Facultade de Enfermería e Podoloxía 92

Filtro por Titulación

☐ Todas as Titulacións
 ☒ 0 Mestrado Universitario en Profesorado de Educación Secun
 ☐ 1 Mestrado Universitario en Profesorado de Educación Secun
 ☐ 2 Mestrado Universitario en Profesorado de Educación Secun
 ☐ 3 Mestrado Universitario en Profesorado de Educación Secun
 ☐ 4 Mestrado Universitario en Profesorado de Educación Secun
 ☐ 5 Mestrado Universitario en Profesorado de Educación Secun

Filtro por Curso

☐ Todos os Cursos
 ☐ Cuarto 560
 ☐ Cuarto Quinto 23
 ☐ Curso Adap. Enx. Téc. Informática 17
 ☐ Primeiro 1777
 ☐ Primeiro Segundo 57
 ☐ Primeiro Segundo Terceiro 37

Figura 9: Filtros.

6.1.4. Búsqueda por rango

Se ha introducido un filtro por rango llamado “**DynamicRangeSlider**”, ver *figura 10*. En este filtro se puede establecer un rango del número de créditos que posee una asignatura.



Figura 10: Filtro de rango para los créditos.

6.2. Ejemplo de búsqueda

En la *figura 11* se puede ver una ejemplo donde se ha buscado una asignatura que contenga la cadena de texto “**linguaxes**”, sea de cualquier curso del grado de informática. El resultado de esta búsqueda devuelve un total de **33 asignaturas que mencionan la palabra “linguaxes”**.

linguaxes

búsqueda: linguaxes X Centro: Facultade de Informática X Titulación: Grao en Enxeñaría Informática X Curso: Todos os Cursos X

Borrar filtros

Filtro por Centros

Buscar centro...

☐ Todos os centros

☒ Facultade de Informática 33

Filtro por Titulación

Buscar titulación...

☐ Todas as Titulacións

☒ Grao en Enxeñaría Informática 33

Mestrado Universitario en Bioinformática para Ciencias da Saúde

Mestrado Universitario en Computación de Altas Prestacións / HI

Mestrado Universitario en Computación de Altas Prestacións / HI

☐ Mestrado Universitario en Enxeñaría Informática (plan 2012) 6

☐ Mestrado Universitario en Xeoinformática (interuniversitario) 7

Filtro por Curso

☒ Todos os Cursos

☒ Cuarto 13

☒ Primeiro 4

☒ Segundo 5

☒ Terceiro 11

33 resultados en 7 ms

Procesamento de Linguaxes

614G01067

Titulación

Grao en Enxeñaría Informática

Centro

Facultade de Informática

Curso

Cuarto

Cuatrimestre

1º cuatrimestre

Tipo

Obrigatoria

Créditos

6

Departamento

Computación

Descripción

Compiladores; tradutores e intérpretes; etapas dun compilador; optimización de código; macroprocesadores. O obxectivo é familiarizar ó alumno co funcionamento dos reconecedores da linguaxe e os compiladores como un caso particular, o entorno no que traballan así como algunhas ferramentas software para a construción dos mesmos. É preciso asumir a característica interdisciplinar da asignatura. Adquirir os coñecementos necesarios para deseñar e implementar as diferentes etapas necesarias para o desenvolvemento dun reconecedor da linguaxe: análise (léxico, sintáctico e semántico) e síntese (xeración de código intermedio, optimización de código e xeración de código obxeto).

Coordinadores

Dafonte Vazquez, Jose Carlos carlos.dafonte@udc.es

Profesores

Arcay Varela, Bernardino bernardino.arcay@udc.es

Dafonte Vazquez, Jose Carlos carlos.dafonte@udc.es

Gomez Garcia, Angel angel.gomez@udc.es

Martinez Perez, Maria maria.martinez@udc.es

Contidos

Tema I. Introdución

1.1 Estructura dun compilador.

1.2 Exemplo das fases dun compilador.

Tema II. Linguaxes e Gramáticas

2.1 Notación e clasificación de Chomsky.

2.2 Gramáticas de contacto libre (GCL) e notación BNF.

3.3 Reduccion e clasificación das gramáticas

Figura 11: Ejemplo de búsqueda con filtros y entrada de texto

15

7. Problemas encontrados

Ha continuación se describen los problemas más relevantes que han sido detectados durante la realización de este proyecto.

7.1. Problemas con Scrappy

A la hora de proceder al crawling, la versión de openssl actual (1.1.1) no es capaz de conectarse con la página de la guía docente. Esta versión se ha introducido durante la realización de la práctica, por lo que a la hora de replicar el resultado en otra máquina, ha sido necesaria la instalación manual en el entorno de la versión 1.0.2 de openssl (como se explica en la *sección 2.1*).

7.2. Problemas con el formato de implementación de la guía docente

El gran problema de este proyecto ha girado entorno a la estructura que sigue la página de la guía docente, puesto que la totalidad de los campos de información carecen de identificadores y por lo tanto ha sido necesario configurar el crawler de una manera muy ad-hoc.

7.3. Problemas con el formato del título de la titulación

A la hora de crear las carpetas/ficheros donde se guardan los resultados de cada asignatura, ha sido necesario sustituir aquellos caracteres especiales que contenían los nombres, ya que se lanzaban errores cuando se intentaban crear dichas carpetas/ficheros con los caracteres especiales.

7.4. Problema cuando no se encuentra ningún resultado en la interfaz

Cuando se realiza una búsqueda desde la interfaz y no se encuentra ningún resultado, no es posible eliminar la búsqueda y realizar otra, es necesario refrescar la página. No se ha detectado el motivo de porque sucede esto. La documentación de `reactiveSearch` no es clara en cuanto a este aspecto.

Referencias

- [1] Elastic. Página Web. <https://www.elastic.co/>.
- [2] Nodejs. Página Web. <https://nodejs.org/es/>.
- [3] Npm. Página Web. <https://www.npmjs.com/>.
- [4] Python. Página Web. <https://www.python.org/>.
- [5] Reactjs. Página Web. <https://reactjs.org/>.
- [6] Anaconda. The most popular python data science platform. Página Web. <https://www.anaconda.com/>.
- [7] Universidade da Coruña. Guías docentes. Página Web. <https://www.udc.es/ensino/guiasdocentes/>.
- [8] Elasticsearch. Kibana. your window into the elastic stack. Página Web. <https://www.elastic.co/products/kibana>.
- [9] ReactiveSearch. A react and react native ui components library for building data-driven apps with elasticsearch. Página Web. <https://github.com/appbaseio/reactivesearch>.
- [10] Scrapy. An open source and collaborative framework for extracting the data you need from websites. in a fast, simple, yet extensible way. Página Web. <https://scrapy.org/>.