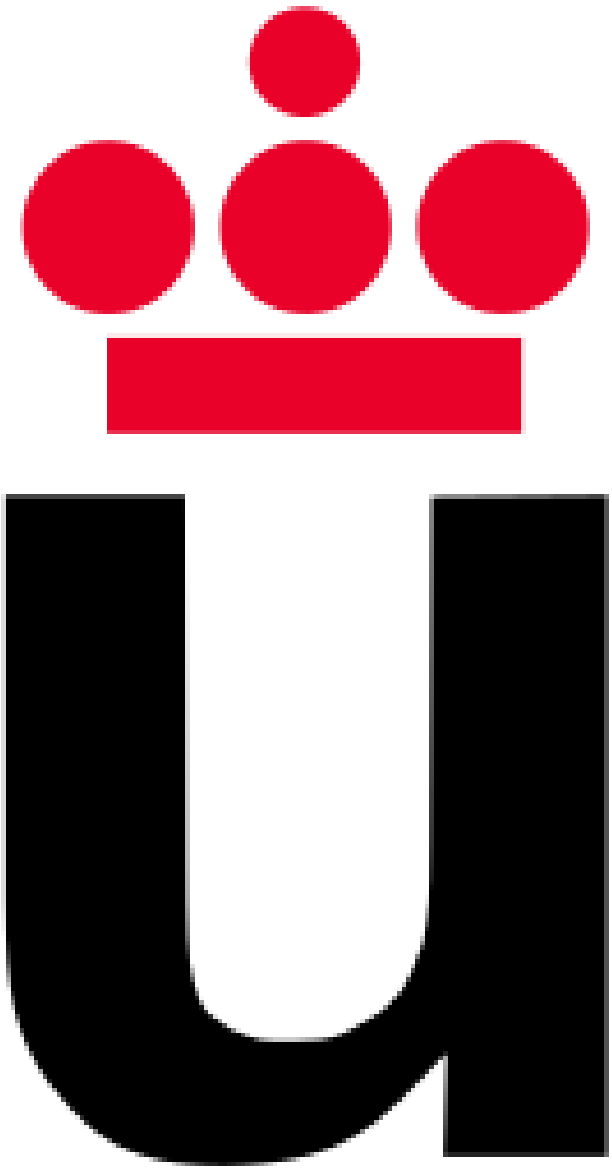
PRÁCTICA OBLIGATORIA: LINKED DATA



Raúl Sanz Cuesta

José Manuel García Llamas

Índice

Parte 1: Resúmenes

Presentación TED Tim Berners-Lee

En esta presentación TED, Tim Berners-Lee empieza contándonos que hace unos años inventó la “World Wide Web” y pidió a la gente que subiera sus documentos en la web para que se pudieran enlazar, pero comenta que esto es solo el principio y que ahora existe otra funcionalidad para la web, la web de datos. Dice que, en muchas organizaciones, es difícil acceder a los datos porque están guardados en bases de datos aisladas y esto impide que fluyan libremente como los documentos en la web.

El objetivo de la web de datos es que toda la información esté enlazada, de forma que los datos de diferentes fuentes puedan conectarse entre sí automáticamente. Tim también nos explica la importancia de usar identificadores universales (URIs) para los elementos de los datos porque si cada concepto tiene su propio URI, es más fácil enlazar la información de distintas fuentes.

Destaca que para que la web de los datos funcione los gobiernos, instituciones y empresas deben publicar sus datos de forma abierta. Esto es importante porque con datos enlazados se pueden descubrir nuevos patrones y relaciones que de otra forma serían casi imposibles de detectar, lo que permitiría revolucionar muchos campos como la ciencia, la medicina y la tecnología entre otros.

Finalmente, termina la charla pidiendo a todos los asistentes que se involucren no solo publicando datos sino también utilizándolos, enlazándolos e innovando con ellos. Quiere que la próxima etapa de la web sea una en la que los datos estén libres y accesibles como ahora lo están los documentos.

Artículo DBpedia

DBpedia es un proyecto creado para extraer información estructurada desde Wikipedia y convertirla en una base de datos en la que los humanos y las máquinas puedan navegar y enlazar datos.

Este proyecto fue creado porque Wikipedia tiene mucho conocimiento en forma de texto escrito, de forma colaborativa por multitud de usuarios y en distintos idiomas, pero este conocimiento no está estructurado para que se pueda utilizar de forma automática.

DBpedia permite extraer hechos y estructuras y convertirlos en datos RDF (Resource Description Framework) que puedan ser consultados y enlazados. El proceso para extraer los datos es el siguiente:

- Utiliza las “infoboxes” de Wikipedia que ya tienen información semiestructurada.
- Los datos extraídos se representan como tripletas RDF con sujeto, predicado y objeto.

- Crea URIs únicas para describir los conceptos concretos.
- Se publican los datos como “Linked Data” accesible en la web.

En cuanto a la arquitectura de extracción, esta utiliza un software llamado “DBpedia Extraction Framework” que reconoce infoboxes, categorías, enlaces internos y otros elementos estructurados. Luego genera varias versiones del dataset en distintos idiomas y, para finalizar, realiza un proceso de limpieza y normalización de los datos extraídos.

Los datos de DBpedia contienen millones de entidades y cientos de millones de tripletas con información sobre personas, lugares, eventos, etcétera. Los datos se extraen de varias ediciones de Wikipedia en distintos idiomas y están organizados usando una ontología (DBpedia Ontology) que define los tipos de entidades y las relaciones entre ellas.

DBpedia Ontology es un conjunto estructurado de clases y propiedades que estandarizan los datos. Permite consultas semánticas complejas usando SPARQL, que es un lenguaje de consulta para datos RDF. Esta ontología se mantiene de forma colaborativa por la comunidad e incluye mapeos entre plantillas de Wikipedia.

En el artículo también se destaca que DBpedia no se encuentra aislada, sino que está enlazada con otras fuentes de datos abiertas como Freebase, YAGO, OpenCyc, GeoNames… Estas conexiones permiten descubrir nuevas relaciones y mejorar la interoperabilidad de los datos. DBpedia forma parte del “Linked Open Data Cloud”, un ecosistema de dataset abiertos conectados entre sí.

Entre las aplicaciones de DBpedia se destaca la exploración de datos enlazados, la mejora de motores de búsqueda, la investigación académica en campos como el procesamiento del lenguaje natural o la minería de datos y el enriquecimiento de otros proyectos que usan DBpedia como fuente.

También se destacan los desafíos que presenta, como la calidad de los datos, ya que hereda errores e inconsistencias de Wikipedia; la necesidad de actualizarse regularmente; la escalabilidad para almacenar esa cantidad masiva de datos y el multilingüismo.

Para finalizar, se habla de las direcciones que tomará el proyecto en el futuro: mejorar el proceso de extracción, incrementar la calidad de los datos mediante validaciones automáticas, integrar mejor información multilingüe y extender las conexiones con otros datasets.

Como conclusión, el artículo afirma que DBpedia ha convertido Wikipedia en una infraestructura clave para el Linked Data y está permitiendo a los usuarios y a las máquinas consultar y utilizar información de Wikipedia de formas nuevas y potentes.

Parte 2: Consulta de datos

Fuente de datos: Wikidata

Descripción: Wikidata es una base de conocimiento libre y abierta que puede ser leída y editada por humanos y máquinas. Sirve como almacenamiento central de datos estructurados para los proyectos de Wikimedia (Wikipedia, Wikivoyage, Wikisource, etc.) y también proporciona soporte a otras aplicaciones más allá del ecosistema Wikimedia.
SPARQL Endpoint: https://query.wikidata.org/
También: https://www.wikidata.org/wiki/Wikidata:Main_Page

Consulta 1: Ideologías de los partidos políticos de los jefes de gobierno actuales, ordenados de forma descendente por el número de países con la misma ideología.

```
SELECT ?ideologiaLabel (COUNT(DISTINCT ?pais) AS ?numeroDePaises)
WHERE {
  ?pais wdt:P6 ?persona .
  ?pais wdt:P31 wd:Q6256 .
  ?persona wdt:P102 ?partido .
  ?partido wdt:P1142 ?ideologia .

  SERVICE wikibase:label { bd:serviceParam wikibase:language "es". }
}
GROUP BY ?ideologiaLabel
ORDER BY DESC(?numeroDePaises)
```

Fichero: wikidataConsulta1.txt

Consulta 2: Sitios declarados Patrimonio de la Humanidad junto con el país al que pertenecen, ordenando los países de forma descendente según el número total de sitios registrados.

```
SELECT ?paisLabel ?sitioLabel WHERE {
  {
    SELECT ?pais (COUNT(?sitio) AS ?totalSitios) WHERE {
      ?sitio wdt:P1435 wd:Q9259 ;
             wdt:P17 ?pais .
    }
    GROUP BY ?pais
  }

  ?sitio wdt:P1435 wd:Q9259 ;
         wdt:P17 ?pais .

  SERVICE wikibase:label { bd:serviceParam wikibase:language "es". }
}
ORDER BY DESC(?totalSitios) ?paisLabel
```

Fichero: wikidataConsulta2.txt

Fuente de datos: Spanish DBpedia

Descripción: Fuente de datos en RDF que extrae y estructura información semántica desde la Wikipedia en español, permitiendo acceder mediante consultas SPARQL a una amplia variedad de datos en un contexto multilingüe y vinculado con otras ediciones de DBpedia.
SPARQL Endpoint: https://es.dbpedia.org/sparql
También: https://es.dbpedia.org/

Consulta 1: Aquellas personas de origen español que contienen una descripción escrita en japonés.

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbr: <http://dbpedia.org/resource/>

SELECT DISTINCT ?name ?description ?urlWikidata WHERE {
  ?person a dbo:Person ;
        foaf:name ?name ;
        dbo:abstract ?description ;
        dbo:birthPlace dbr:Spain ;
        owl:sameAs ?urlWikidata .
  FILTER (STRSTARTS(STR(?urlWikidata), "http://www.wikidata.org/entity/"))
  FILTER (lang(?name) = "en")
  FILTER (lang(?description) = "ja")
}
ORDER BY ?name
```

Fichero: spanishDbpediaConsulta1.txt

Consulta 2: Obtiene todas aquellas personas relacionadas con Japón y que practican algún deporte mostrando su URI, Nombre y Deporte. Limitado a un máximo de 100 filas.

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT DISTINCT
  ?persona AS ?Link
  (STR(?nombre) AS ?Nombre)
  (REPLACE(STRAFTER(STR(?deporte), "resource/"), "_", " ") AS ?Deporte)
WHERE {
  ?persona a foaf:Person .
  ?persona ?p ?tema .
  FILTER (?p IN (dct:subject, skos:subject)) .
  FILTER (
    CONTAINS(LCASE(STR(?tema)), "japon") ||
    CONTAINS(LCASE(STR(?tema)), "japón")
  )

  OPTIONAL { ?persona foaf:name ?nombre . }
  ?persona dbo:sport ?deporte .
```

```
}
LIMIT 100
```

Fichero: spanishDbpediaConsulta2.txt

Fuente de datos: Nobel Prize Linked Data

Descripción: Fuente de datos que contiene información estructurada sobre los premios Nobel, sus categorías, ganadores, nacionalidades, género y otros atributos relacionados.
SPARQL Endpoint: http://data.nobelprize.org/sparql
También: https://www.nobelprize.org/

Consulta 1: Número de personas que han recibido el Premio Nobel, clasificadas por género y categoría, excluyendo organizaciones y entradas sin género declarado. Ordenado de forma ascendente por el número total de premios por categoría y con etiquetas traducidas al español.

```
SELECT
  ?categoriaNombreES
  ?generoES
  (STR(COUNT(DISTINCT ?persona)) AS ?totalGanadores)
WHERE {
  {
    # Subconsulta para calcular total por categoría
    SELECT ?categoria (COUNT(DISTINCT ?persona) AS
?totalPremiosCategoria) WHERE {
      ?premio a <http://data.nobelprize.org/terms/LaureateAward> ;
            <http://data.nobelprize.org/terms/laureate> ?persona ;
            <http://data.nobelprize.org/terms/category> ?categoria .
    }
    GROUP BY ?categoria
  }

  # Consulta principal
  ?premio a <http://data.nobelprize.org/terms/LaureateAward> ;
        <http://data.nobelprize.org/terms/laureate> ?persona ;
        <http://data.nobelprize.org/terms/category> ?categoria .

  ?persona <http://xmlns.com/foaf/0.1/gender> ?genero .

  # Extraer y traducir nombre de categoría
  BIND(STRAFTER(STR(?categoria), "category/") AS ?catText)
  BIND(REPLACE(?catText, "_", " ") AS ?catNombre)
  BIND(
    IF(?catNombre = "Chemistry", "Química",
    IF(?catNombre = "Physics", "Física",
    IF(?catNombre = "Literature", "Literatura",
    IF(?catNombre = "Peace", "Paz",
    IF(?catNombre = "Economic Sciences", "Ciencias Económicas",
```

```
    IF(?catNombre = "Physiology or Medicine", "Fisiología o Medicina",
?catNombre)))))) AS ?categoriaNombreES)

  # Traducir género
  BIND(
    IF(?genero = "male", "hombre",
    IF(?genero = "female", "mujer", ?genero)) AS ?generoES)
}
GROUP BY ?categoriaNombreES ?generoES ?totalPremiosCategoria
ORDER BY ASC(?totalPremiosCategoria) ?categoriaNombreES ?generoES
```

Fichero: nobelPrizeConsulta1.txt

Consulta 2: Número total de premios Nobel otorgados en cada década y cuántos de ellos fueron concedidos a organizaciones (en lugar de personas), sin premios duplicados.

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?decada
       (STR(?totalPremiosRaw) AS ?totalPremios)
       (STR(?premiosOrganizacionRaw) AS ?premiosOrganizacion)
WHERE {
  {
    SELECT ?decada (COUNT(DISTINCT ?premio) AS ?totalPremiosRaw)
WHERE {
      ?premio a <http://data.nobelprize.org/terms/LaureateAward> ;
            <http://data.nobelprize.org/terms/year> ?anio .
      BIND(STR(?anio) AS ?anioStr)
      BIND(CONCAT("Década de ", SUBSTR(?anioStr, 1, 3), "0") AS ?decada)
    }
    GROUP BY ?decada
  }

  {
    SELECT ?decada (COUNT(DISTINCT ?premio) AS
?premiosOrganizacionRaw) WHERE {
      ?premio a <http://data.nobelprize.org/terms/LaureateAward> ;
            <http://data.nobelprize.org/terms/year> ?anio ;
            <http://data.nobelprize.org/terms/laureate> ?ganador .
      FILTER NOT EXISTS { ?ganador foaf:gender ?genero }
      BIND(STR(?anio) AS ?anioStr)
      BIND(CONCAT("Década de ", SUBSTR(?anioStr, 1, 3), "0") AS ?decada)
    }
    GROUP BY ?decada
  }
}
ORDER BY ?decada
```

Fichero: nobelPrizeConsulta2.txt

Fuente de datos: datos.gob.es Endpoint

Descripción: Fuente de datos del gobierno de España que proporciona información estructurada sobre la organización territorial de España, incluyendo comunidades autónomas, provincias, municipios y otras divisiones administrativas, así como sus relaciones jerárquicas.
SPARQL Endpoint: https://datos.gob.es/es/sparql
También: https://datos.gob.es/es/

Consulta 1: Lista de comunidades autónomas ordenadas de forma ascendente por su número de provincias y alfabéticamente.

```
SELECT ?comunidadLabel ?provinciaLabel (IF(?provinciaOrden =
?primeraProvincia, STR(?numProvincias), "") AS ?numProvinciasTexto)
WHERE {
  {
    SELECT ?comunidad ?comunidadLabel (COUNT(DISTINCT ?provincia)
AS ?numProvincias)
    WHERE {
      ?provincia <http://vocab.linkeddata.es/datosabiertos/def/sector-
publico/territorio#autonomia> ?comunidad .
      ?provincia <http://www.w3.org/2000/01/rdf-schema#label>
?provinciaLabel .
      ?comunidad <http://www.w3.org/2000/01/rdf-schema#label>
?comunidadLabel .
    }
    GROUP BY ?comunidad ?comunidadLabel
  }

  ?provincia <http://vocab.linkeddata.es/datosabiertos/def/sector-
publico/territorio#autonomia> ?comunidad .
  ?provincia <http://www.w3.org/2000/01/rdf-schema#label> ?provinciaLabel .

  # Normalización para orden alfabético de provincias
  BIND(REPLACE(LCASE(?provinciaLabel), "á", "a") AS ?provinciaOrden)

  {
    SELECT ?comunidad (MIN(REPLACE(LCASE(?provinciaLabel), "á", "a"))
AS ?primeraProvincia)
    WHERE {
      ?provincia <http://vocab.linkeddata.es/datosabiertos/def/sector-
publico/territorio#autonomia> ?comunidad .
      ?provincia <http://www.w3.org/2000/01/rdf-schema#label>
?provinciaLabel .
    }
    GROUP BY ?comunidad
  }
}
ORDER BY DESC(?numProvincias) ?comunidadLabel ?provinciaOrden
```

Fichero: datosGobConsulta1.txt

Consulta 2: Temas de los datasets publicados, ordenados de forma ascendente según el número total de publicaciones, indicando también qué

organización ha publicado más datasets sobre ese tema y cuántos ha publicado.

```
PREFIX dcat: <http://www.w3.org/ns/dcat#>
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?tema (STR(?numDatasetsTema) AS ?totalDatasetsTexto)
?organizacionMax (STR(?numDatasetsOrg) AS ?orgDatasetsTexto)
WHERE {
 {
  SELECT ?tema (COUNT(DISTINCT ?dataset) AS ?numDatasetsTema)
  WHERE {
   ?dataset a dcat:Dataset ;
          dcat:theme ?tema .
  }
  GROUP BY ?tema
 }

 {
  SELECT ?tema ?organizacionMax (COUNT(?dataset) AS
?numDatasetsOrg)
  WHERE {
   ?dataset a dcat:Dataset ;
          dcat:theme ?tema ;
          dct:publisher ?organizacionMax .
  }
  GROUP BY ?tema ?organizacionMax
 }

 {
  SELECT ?tema (MAX(?cuenta) AS ?maxCuenta)
  WHERE {
   {
    SELECT ?tema ?organizacion (COUNT(?dataset) AS ?cuenta)
    WHERE {
     ?dataset a dcat:Dataset ;
            dcat:theme ?tema ;
            dct:publisher ?organizacion .
    }
    GROUP BY ?tema ?organizacion
   }
  }
  GROUP BY ?tema
 }

 FILTER(?numDatasetsOrg = ?maxCuenta)
}
ORDER BY DESC(xsd:integer(?totalDatasetsTexto))
```

Fichero: datosGobConsulta2.txt

Fuente de datos: UniProt

Descripción: Fuente de datos de referencia sobre proteínas. Su objetivo es ofrecer una fuente centralizada y de alta calidad para la investigación en biología molecular y bioinformática. UniProt integra datos de múltiples recursos, incluyendo UniProtKB, UniRef y UniParc.
SPARQL Endpoint: https://sparql.uniprot.org/sparql
También: https://www.uniprot.org/

Consulta 1: Proteínas más recomendadas para la especie homo sapiens y una explicación de su función.

```
PREFIX up: <http://purl.uniprot.org/core/>
PREFIX taxon: <http://purl.uniprot.org/taxonomy/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?proteina ?nombre ?explicacion WHERE {
  ?proteina a up:Protein ;
          up:organism taxon:9606 ;
          up:recommendedName ?rec;
          up:annotation ?anotacion .

  ?rec up:fullName ?nombre .

  ?anotacion a up:Function_Annotation ;
                    rdfs:comment ?explicacion .
}
```

Fichero: uniprotConsulta1.txt

Consulta 2: Proteinas del gato relacionadas con el fallecimiento y una explicación de estas.

```
PREFIX up: <http://purl.uniprot.org/core/>
PREFIX taxon: <http://purl.uniprot.org/taxonomy/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
SELECT ?proteinaTexto ?genTexto ?comentarioTexto
WHERE {
  ?protein a up:Protein;
          up:organism taxon:9685;
          rdfs:label ?nombreProteina;
          up:encodedBy ?gene;
          up:annotation ?nota.
  ?gene skos:prefLabel ?nombreGen.
  ?nota a up:Disease_Annotation;
        rdfs:comment ?comentario.
  BIND(STR(?nombreProteina) AS ?proteinaTexto)
  BIND(STR(?nombreGen) AS ?genTexto)
  BIND(STR(?comentario) AS ?comentarioTexto)
}
LIMIT 50
```

Fichero: uniprotConsulta2.txt

Parte 3: Aplicación de uso de Linked Data

Introducción

Aprovechando los datos contenidos tanto en BDpedia como en Wikidata, hemos decidido hacer una pequeña aplicación que nos muestra listados de actores según nuestras preferencias mediante el uso de filtros. En estos filtros, mostraremos características que podrían interesar a la hora de buscar cierto tipo de actores. Si el usuario quiere descubrir más información de un actor o actriz en concreto, nuestra aplicación también muestra una ficha con sus datos personales, una foto suya, películas en las que ha participado, una breve descripción de este y los premios que ha recibido. Para complementar la explicación, hemos decidido adjuntar un vídeo en el que mostramos el funcionamiento de nuestra aplicación en detalle.

Como observación, hemos visto que las dimensiones de las ventanas son distintas en Windows y en Mac, así que hemos decidido adaptarlas para que se pudiera mostrar el contenido correctamente en ambos sistemas operativos. El ejecutable también es distinto, pero hemos decidido incluir solo el archivo ejecutable de Windows porque el de Mac era demasiado pesado.

Objetivo

Nuestro objetivo es que el usuario pueda conocer a los diversos actores que pertenecen ya sea a su género de películas o país favorito y que además pueda filtrarlos por su edad. Aparte de estas características básicas, permitimos al usuario profundizar en datos más específicos de los actores que quiera para que pueda ver una foto de ellos si solo recordaba el nombre, películas en las que han formado parte y que a lo mejor no lo sabía y premios o reconocimientos que hayan podido recibir, aumentando así su conocimiento cinematográfico.

Prototipo y funcionamiento

Al abrir la aplicación tenemos varios filtros que podremos usar para buscar nuestros posibles actores de interés. Para la prueba hemos usado únicamente tres filtros, de los cuales podemos seleccionar uno o varios simultáneamente. Una vez seleccionados los filtros que queramos, pulsaríamos el botón “Buscar” para que nos muestre la lista de actores que cumplen con los requisitos que hemos seleccionado. Debido a que es posible que la búsqueda tarde algo de tiempo, hemos decidido poner una ventana emergente para informar al usuario de que la aplicación está buscando los actores y sepa que está funcionando correctamente, evitando que haga peticiones innecesarias por pulsar repetidas veces el botón “Buscar”.

Una vez aparece el listado, el usuario puede seleccionar el nombre de cualquier actor o actriz para poder ver más información. Al seleccionarlo, aparecerá el

nombre debajo de la etiqueta “Buscar más información de:”, para que el usuario pueda comprobar que ha seleccionado el actor correcto, y al pulsar el botón “Detalles” le mostrará la ficha con los datos previamente mencionados de ese actor en una nueva ventana. La información se muestra de forma similar a la Wikipedia, pero centrándonos en los datos más importantes para nuestros usuarios.

Teníamos otras ideas para la implementación de la aplicación, pero por falta de tiempo y para favorecer la usabilidad y simpleza de esta hemos decidido desecharlas. Estos eran algunos de los cambios que pensábamos implementar:

- Actualmente el listado de actores aparece únicamente con sus nombres, pero habíamos pensado que tuviera un formato más visual y que apareciera también la foto de cada uno al lado de sus nombres, para que los usuarios pudieran identificar más fácilmente cada uno de los actores.
- La búsqueda por teclado en el buscador tampoco está implementada en la aplicación. Era una idea para agilizar la búsqueda de actores en específico, permitiendo que el usuario tuviera más de una forma de buscar y encontrar a un actor que le interesase.