



**UNIVERSIDAD TÉCNICA PARTICULAR
DE LOJA ÁREA TÉCNICA**

**SISTEMAS BASADO EN
CONOCIMIENTOS**

LINKED DATA APLICADA AL CAMPO TURÍSTICO

Docente: JANNETH CHICAIZA ESPINOSA, Msc.

Integrantes:

- Cuenca Paúl
- Mena Andrea

17/03/2019

Contenido

1	DOCUMENTACIÓN DE GENERACIÓN DE DATOS RDF	3
1.2.	Diseño de modelo entidad relación y modelo semiformal de la ontología.	3
1.3.	Creación del modelo ontológico en la herramienta Protege.	4
1.4.	Creación del modelo (clases, propiedades) e instanciación con los datos del csv principal, a través de la librería Jena en Java.	6
1.5.	Almacenamiento de los datos RDF en la herramienta de OpenLinkVirtuoso.	9
2	DOCUMENTACIÓN DE DESARROLLO DE LA APLICACIÓN	10
2.1.	PROCESO.....	10
<input type="checkbox"/>	Nombre.....	10
<input type="checkbox"/>	Ciudad.....	11
<input type="checkbox"/>	Actividad.....	11
2.2.	Anexos de funcionamiento de la aplicación.....	12

1 DOCUMENTACIÓN DE GENERACIÓN DE DATOS RDF

PROCESO

1.1. Limpieza de datos en el formato XMLS.

(Data obtenida del Ministerio de Turismo de Ecuador). En este paso se realizó en primera instancia la limpieza de algunos datos de información principal. (Se eliminó algunas columnas y registros innecesarios e irrelevantes para su posterior uso). Se procedió a convertir toda la data en formato CSV (delimitada por punto y coma), para facilitar la lectura de todos los datos.

Especificación de dataset final usado para el proyecto.

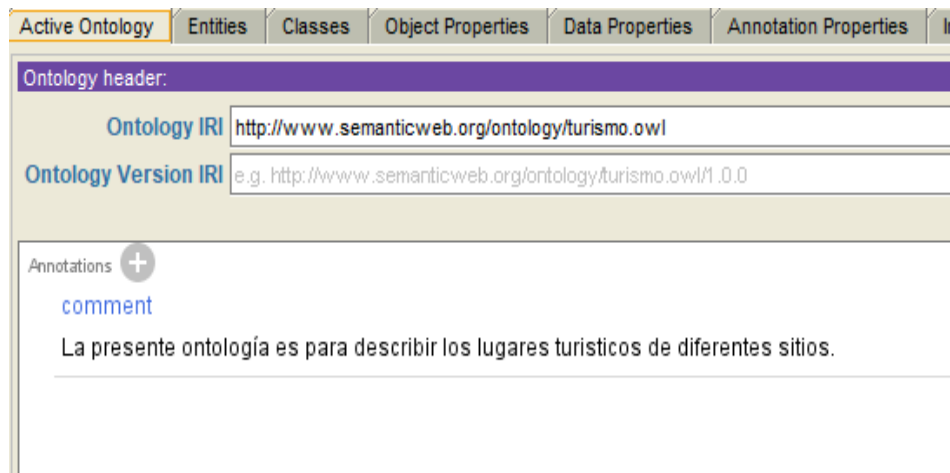
- ✚ **RUC** (Identificación única)
- ✚ **Nombre comercial** (Nombre del sitio)
- ✚ **Actividad** (A qué se dedica el sitio)
- ✚ **Clasificación** (Describe a qué clasificación dentro de la actividad a la que pertenece)
- ✚ **Categoría** (Específica a la categoría que pertenece, 1, 2, 3, 4, 5, 6 estrellas)
- ✚ **Local** (Describe el tipo de local, principal/sucursal)
- ✚ **Provincia** (Provincia a la cual pertenece)
- ✚ **Cantón** (Cantón a la cual pertenece)
- ✚ **Referencia** (Describimos una pequeña referencia respecto a su ubicación)
- ✚ **Dirección** (Describe la ubicación)
- ✚ **TelefonoCelular** (Número de Teléfono celular por el cual se comunicará)
- ✚ **Email** (Describe la dirección de correo electrónico del sitio)
- ✚ **Web** (Describe la dirección o url de la página web)
- ✚ **HABITACIONES** (El número de habitaciones que tiene el sitio turístico)
- ✚ **CAMAS** (El número de camas que tiene el sitio turístico)
- ✚ **MESAS** (El número de mesas que tiene el sitio turístico)
- ✚ **Latitud** (Dato de ubicación)
- ✚ **Longitud** (Dato de ubicación)
- ✚ **AdministracionZonal** (Describe el nombre de la administración zonal a la que pertenece a un distrito metropolitano)
- ✚ **SectorTuristico** (Describe el sector exacto de donde se encuentra el sitio turístico)

1.2. Diseño de modelo entidad relación y modelo semiformal de la ontología.

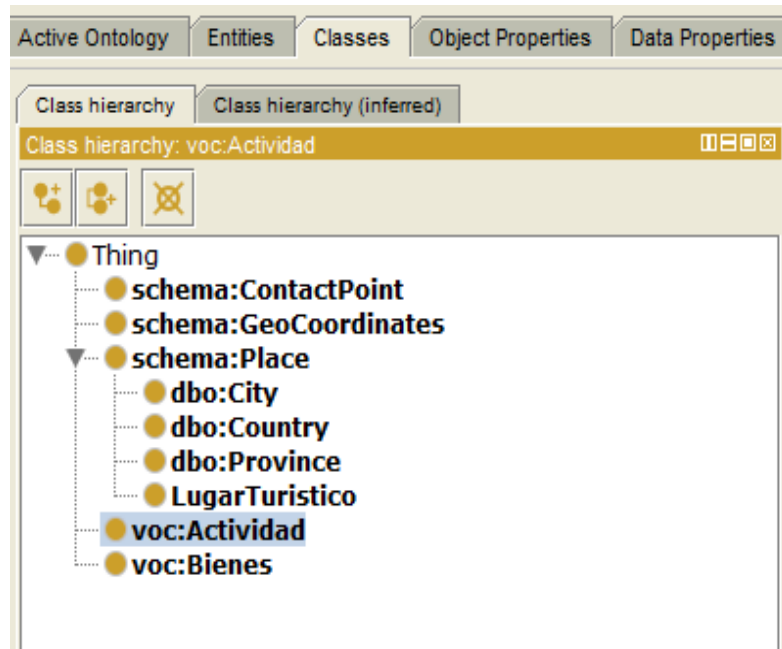
- a. Se generó un modelo de entidad con base en las columnas que se especificó utilizar y se determinó las posibles relaciones existentes entre ellas.
- b. Posteriormente se creó la especificación del modelo semiformal con el uso de los vocabularios: schema, dbpedia, foaf, rdfs. Además, se hizo un vocabulario propio para las propiedades y clases que no fueron posibles reutilizar.

1.3. Creación del modelo ontológico en la herramienta Protege.

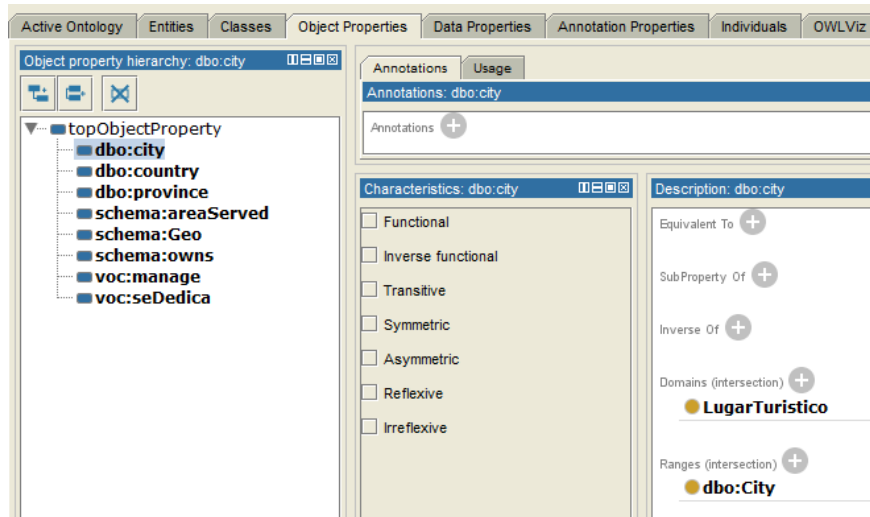
- a. Como inicio del modelo en esta herramienta se define una IRI específica para la ontología, además, se puede describir brevemente la ontología.



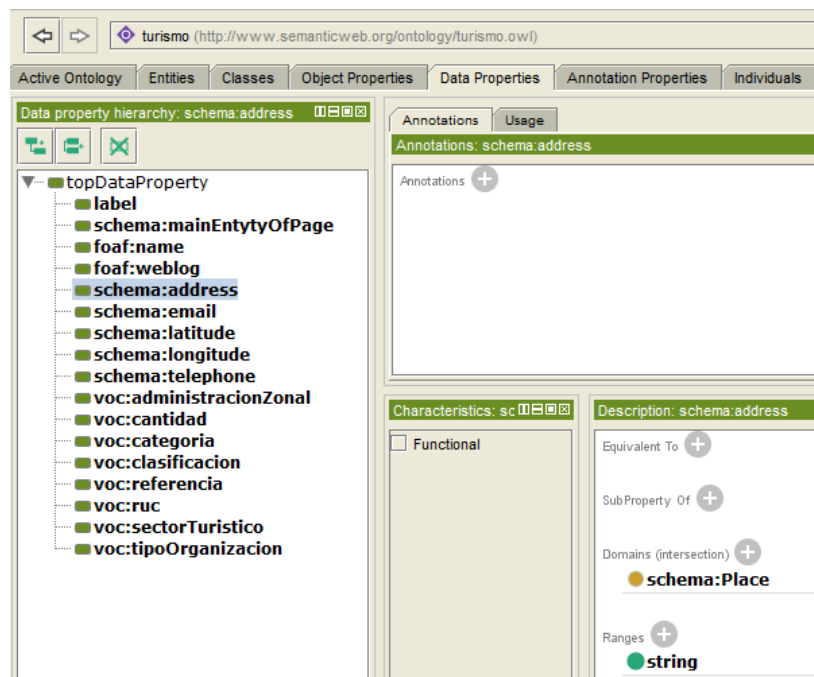
- b. Luego, se crean una a una las clases y se determina las subclases existentes y las relaciones que ellas tenga.



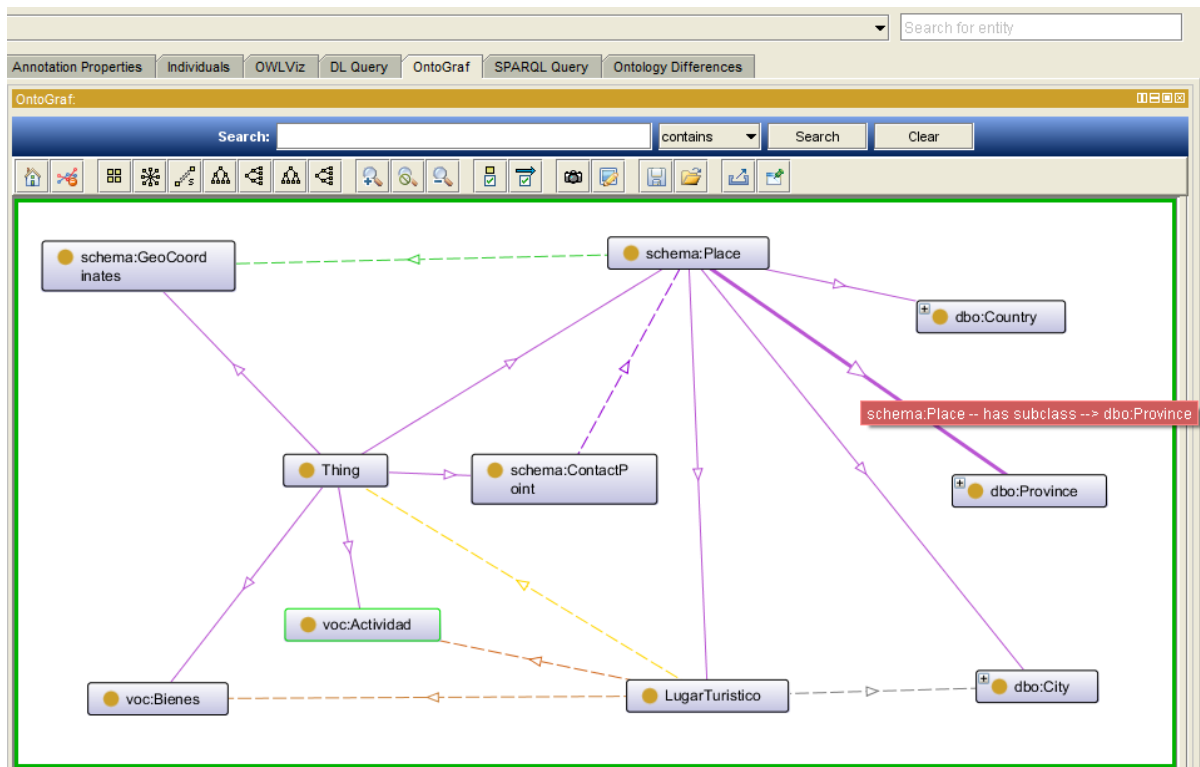
- c. A continuación, se crean todas las propiedades que tienen como rango otra clase, especificando tanto su dominio como rango de acuerdo las clases previamente creadas.



- d. Seguido a ello se definen las propiedades que tienen como rango los literales en las cuales se determina como dominio la clase correspondiente y como rango el tipo de dato al que pertenezca.



- e. Finalmente se puede visualizar el grafo de las clases y las propiedades entre ellas.



1.4. Creación del modelo (clases, propiedades) e instanciación con los datos del csv principal, a través de la librería Jena en Java.

Siguiendo el ejemplo que se nos facilitó por parte de la docente en el aula, se realizó el diseño de un modelo ontológico para la creación del archivo RDF que se utilizara para su futuro almacenamiento en algún repositorio y BD, siempre y cuando soporten la estructura de datos en RDF (**sujeto - predicado - objeto**).

A continuación se detalla en pasos lo que se realizó en Netbeans:

- ✚ Se creó un proyecto en Java, la cual consta de 2 clases referentes al proyecto. 1 interfaz para la aplicación.
 - ✚ La clase **LugarTuris.java** se la creó con el fin de poder tener una estructura de objeto y así facilitar el almacenamiento y lectura del archivo CSV.
 - ✚ La clase **JenaTurismo.java** aquí se creó todos los métodos para la lectura del archivo CSV, creación del modelo ontológico e instanciación de datos y su objetivo principal que es la creación del archivo en formato RDF.
- La primera parte de la clase se la creó para la lectura del archivo CSV y el cual se lo va almacenando en un ArrayList de tipo **Lugar Turis**, donde constan todos los campos necesarios para el posterior trabajo.

```

public class JenaTurismo {
    public static final String SEPARATOR = ";";
    public static final String SEPARATOR2 = ",";
    public static final ArrayList<LugarTuris> lugares= new ArrayList<LugarTuris>();

    public static void main(String[] args) throws FileNotFoundException, IOException {
        File fi = new File("C:\\Users\\Florcita\\Documents\\NetBeansProjects\\sbcPractica\\src\\csvcsv2.csv");
        StringTokenizer st;
        Scanner entrada = null;
        String sCadena;
        try {
            entrada = new Scanner(fi);
            while (entrada.hasNext()) {
                LugarTuris p = new LugarTuris();
                sCadena = entrada.nextLine();
                // System.out.println(sCadena);
                st = new StringTokenizer(sCadena, ";");
                while (st.hasMoreTokens()) {
                    p.setRUC(st.nextToken());
                    p.setLegalName(st.nextToken());
                    p.setActividad(st.nextToken());
                    p.setClasificacion(st.nextToken());
                    p.setCategoria(st.nextToken());
                    p.setTipoLugar(st.nextToken());
                    p.setProvincia(st.nextToken());
                    p.setCiudad(st.nextToken());
                    p.setReferencia(st.nextToken());
                    p.setDireccion(st.nextToken());
                    p.setTelefono(st.nextToken());
                    p.setEmail(st.nextToken());
                    p.setWebBlog(st.nextToken());
                    p.setCantidadHabitacion(st.nextToken());
                    p.setCantidadCama(st.nextToken());
                    p.setCantidadMesa(st.nextToken());
                    p.setLatitud(st.nextToken());
                    p.setLongitud(st.nextToken());
                    p.setAdministracionZonal(st.nextToken());
                    p.setSectorTuristico(st.nextToken());
                }
                lugares.add(p);
            }
        } catch (FileNotFoundException e) {
    }
}

```

- Posteriormente se crea el modelo vacío para luego y se crea el archivo donde se almacenará toda la información que será exportada en formato RDF.
- Luego se declara todos los vocabularios que se va a utilizar en el modelo, teniendo en cuenta que se creó nuestros propios vocabularios en algunos campos de la data.
- Se crearon todas las URIS correspondientes para los conceptos(**clases**), propiedades que se usarán en modelo de la ontología.

```

//Create an empty model
Model model = ModelFactory.createDefaultModel();
File f = new File("C:\\Users\\Florcita\\Documents\\NetBeansProjects\\sbcPractica\\src\\archivoRDF.rdf");
FileOutputStream os = new FileOutputStream(f);
System.out.println("hola estoy x aqui");
//Fijar Prefijo para URI base de dos datos a crear
String voc = "http://turismo.org/voc/";
model.setNsPrefix("voc", voc);

String schema = "http://schema.org/";
model.setNsPrefix("schema", schema);

String dbo = "http://dbpedia.org/ontology/";
model.setNsPrefix("dbo", dbo);

String dbr = "http://dbpedia.org/resource/";
model.setNsPrefix("dbr", dbr);

String dataPrefix = "http://example.org/data/";
model.setNsPrefix("data", dataPrefix);

//Fijar prefijos de vocabularios incorporados en Jena
String foaf = "http://xmlns.com/foaf/0.1/";
model.setNsPrefix("foaf", foaf);

// en el caso de vocabularios externos (no incorporados en Jena) se debe crear un modelo
Model dboModel = ModelFactory.createDefaultModel(); // modelo para la ontología
//uris y valores para conceptos
String lugarTuristicoClassURI = voc+"LugarTuristico";
String CityClassURI = dbo+"City";
String ProvinceClassURI = dbo+"Province";
String CountryClassURI = dbo+"Country";
String ActividadClassURI = voc+"Actividad";
String BienesClassURI = voc+"Bienes";
String PlaceClassURI = schema+"Place";
String ContactPointClassURI = schema+"ContactPoint";
String GeoCoordinatesClassURI = schema+"GeoCoordinates";

//uris y valores para propiedades
String managePropertiURI = voc+"manage";
String rucPropertiURI = voc+"ruc";
String activityPropertiURI = voc+"actividad";
String adminZonalPropertiURI = voc+"administracionZonal";
String sectorTurisPropertiURI = voc+"sectorTuristico";
String cityPropertiURI = dbo+"city";
String referenciaPropertiURI = voc+"referencia";

```

- Se procedió a modelar la ontología conforme la relación de los datos en el archivo que se examinó y la escritura de toda la información estructurada en RDF en un archivo que se exportará.


```

String var="LugarTurismo";
for (int i = 0; i < lugares.size(); i++) {
    //Creando el modelo de
    Resource LugarTurismo = model.createResource(LugarTurismoClassURI);
    StmtIterator iter = model.listStatements();
    // print out the predicate, subject and object of each statement
    while (iter.hasNext()) {
        Statement stmt = iter.nextStatement(); // get next statement
        Resource subject = stmt.getSubject(); // get the subject
        Property predicate = stmt.getPredicate(); // get the predicate
        RDFNode object = stmt.getObject(); // get the object
        System.out.print(subject.toString());
        System.out.print(" " + predicate.toString() + " ");
        if (object instanceof Resource) {
            System.out.print(object.toString());
        } else {
            // object is a literal
            System.out.print(" \" " + object.toString() + "\"");
        }
        System.out.println(" .");
    }
    Resource Actividad = model.createResource(ActividadClassURI);
    // now write the model in XML form to a file
    System.out.println("MODELO RDF-----");
    model.write(System.out, "RDF/XML-ABBREV");
    Resource Bienes = model.createResource(BienesClassURI);
    // Save to a file
    RDFWriter writer = model.getWriter("RDF/XML"); //RDF/XML
    writer.write(model, os, "");
    Resource City = model.createResource(CityClassURI);
    //Cerrar modelos
    dboModel.close();
    Resource Province = model.createResource(ProvinceClassURI);
    model.close();
    .addProperty(RolType, model.createResource(ProvinceClassURI));
    .addProperty(country, dboModel.getResource(dbr + "Ecuador"));
    Resource ContacPoint = model.createResource(ContactPointClassURI + "/" + "ContactPoint" + i);
    .addProperty(email, lugares.get(i).getEmail());
    .addProperty(telephone, lugares.get(i).getTelefono());
    Resource GeoCoordinates = model.createResource(GeoCoordinatesClassURI + "/" + "GeoCoordinates" + i);
    .addProperty(latitude, schema + lugares.get(i).getLatitud());
    .addProperty(longitude, schema + lugares.get(i).getLongitud());
    System.out.println("¡llegue aqui!");
}

```

1.5. Almacenamiento de los datos RDF en la herramienta de OpenLinkVirtuoso.

Se instaló la aplicación OpenLinkVirtuoso, en el cual nos conectamos con credenciales de **usuario:dba** | **contraseña:dba** en primera nos vamos a **Database** y en us sub menú **Interactive SQL**, en esta zona ingresamos los siguientes comando para:

- Creación del grafo donde se abarcará toda la ontología.
SPARQL CREATE GRAPH <http://data.utpl.edu.ec/ontologyturismo>;
- Carga a partir desde un archivo de tripletas en formato .rdf.
DB.DBA.RDF_LOAD_RDFXML_MT (file_to_string_output ('C:/Program Files/OpenLink Software/Virtuoso OpenSource 7.2/database/archivoRDF2.rdf'), ", 'http://data.utpl.edu.ec/LugaresTuristicos');

Luego en la parte de **Linked Data** se pueden realizar todas las consultas necesarias directas a los datos cargados anteriormente.

2 DOCUMENTACIÓN DE DESARROLLO DE LA APLICACIÓN

La interfaz consta de un panel que posee como primer elemento muestra un menú con las opciones de búsqueda, es decir se puede seleccionar el criterio por el cual se quiere obtener información de los lugares turísticos. A continuación, de acuerdo a lo seleccionado se escribe el nombre del lugar o la ciudad que se quiere obtener información, o bien, si se trata del criterio de la actividad se mostrará un menú de opciones específicas. Los resultados a presentarse serán observables en una tabla.

2.1. PROCESO

❑ Nuestra aplicación se basa en un buscador básico, donde podremos explorar la data por 3 campos en específico. Todo esto se lo puede hacer mediante unas consultas realizadas en **SPARQL** las cuales se ejecutan en la clase **GUlturismo**

❑ **Nombre.** - Con este campo de búsqueda podremos tener acceso al:

- ❑ Nombre del lugar turístico
- ❑ La actividad a la que se dedican
- ❑ La clasificación de acuerdo a su actividad
- ❑ La categoría a la que pertenece (1, 2, 3, 4, 5, 6 estrellas)
- ❑ El tipo de local (principal y sucursal)
- ❑ La dirección del sitio

```
prefix dbo: <http://dbpedia.org/ontology/>
prefix voc: <http://turismo.org/voc/>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix schema: <http://schema.org/>
select distinct ?s ?o ?p ?o2 ?o3 ?o4
from <http://data.utpl.edu.ec/LugaresTuristicos>
where{
?a foaf:name "DON FRANCISCO EL SABOR ORIGINAL".
?a rdf:type ?p.
?a voc:actividad ?y.
?y rdfs:label ?o; voc:clasificacion ?o2; voc:categoria ?o3.
?a voc:tipoOrganizacion ?o4; schema:address ?o5
}
```

- ❑ **Ciudad.** - Con este campo de búsqueda podremos tener acceso a los lugares turísticos que pertenecen a esta ciudad:

- ☐ Nombre del lugar turístico
- ☐ Provincia a la que pertenece
- ☐ País al que pertenece la provincia
- ☐ Latitud del sitio
- ☐ Longitud del sitio

```
prefix dbo: <http://dbpedia.org/ontology/>
prefix voc: <http://turismo.org/voc/>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix schema: <http://schema.org/>
select distinct ?s ?o ?o2 ?o3 ?o5 ?o6
from <http://data.utpl.edu.ec/LugaresTuristicos>
where{
?a dbo:city <http://dbpedia.org/ontology/Quito>.
?a foaf:name ?o.
?s dbo:province ?o2.
?o2 dbo:country ?o3.
?a schema:geo ?o4.
?o4 schema:latitude ?o5; schema:longitude ?o6
}
```

- ❑ **Actividad.** - Con este campo de búsqueda podremos tener acceso a los lugares que se dediquen a esta actividad. Y tendemos acceso a más información relevante del sitio.

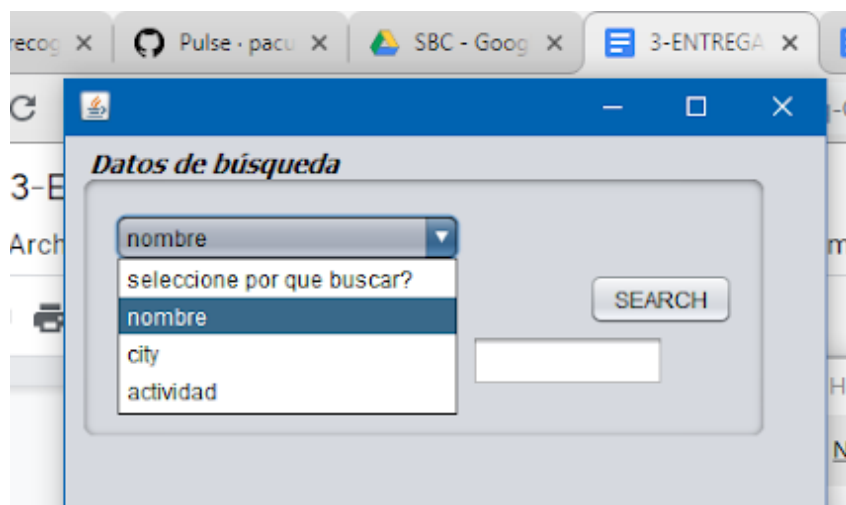
- ☐ Nombre del lugar turístico
- ☐ Número de teléfono
- ☐ Email de la propietaria
- ☐ Página web del sitio
- ☐ Cantidad de mesas (si fuese restaurante)
- ☐ Cantidad de camas (si fuese hostel u hotel)
- ☐ Cantidad de habitaciones (si fuese hostel u hotel)

```

prefix dbo: <http://dbpedia.org/ontology/>
prefix voc: <http://turismo.org/voc/>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix schema: <http://schema.org/>
select distinct ?p ?o1 ?o2 ?o3 ?o5 ?o6 ?o7
from <http://data.utpl.edu.ec/LugaresTuristicos>
where{
  ?a voc:actividad ?s.
  ?s rdfs:label "Alimentos y Bebidas".
  ?a foaf:name ?p.
  ?a schema:areaServed ?o.
  ?o schema:telephone ?o1; schema:email ?o2.
  ?a foaf:weblog ?o3.
  ?a schema:owns ?o4.
  ?o4 voc:cantMesa ?o5; voc:cantCama ?o6 ; voc:cantHabitacion ?o7.
}

```

2.2. Anexos de funcionamiento de la aplicación



Búsqueda por city "Quito" y su resultado

The image shows a search application interface. At the top, a window titled "Datos de búsqueda" (Search Data) contains a dropdown menu with "city" selected and a "SEARCH" button. Below this, the text "Ciudad:" (City:) is followed by a text box containing "Quito".

Below the search window, a larger window titled "Data" displays a table of search results. The table has five columns: "Lugar-Turístico" (Tourist Place), "Provincia" (Province), "País" (Country), "Latitud" (Latitude), and "Longitud" (Longitude). The first row of data is "DON FRANCISCO E...", which corresponds to the search query "DON FRANCISCO EL SABOR ORIGINAL".

Lugar-Turístico	Provincia	País	Latitud	Longitud
DON FRANCISCO E...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
HELADERIA LOS A...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
\RESTAURANTE \...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
SWEET & COFFEE	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
M&M TRAVEL GROUP	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
BURGER KING	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
LUJOR	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
LA TRIBU TAINA	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
SELINA EXPLORE Q...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
KARAOKE DIVER FULL	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
CARLOS LUIS	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
BOUTIQUE + ARTE ...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
LA HUECA DE CANT...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
VANSERVICE INTER...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
ARTE Y VIDA	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/0	http://schema.org/0
EL PORTAL	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
LA ESTANCIA ESME...	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
CHULLA & CHULLA	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/0	http://schema.org/0
EL CONO GIGANTE	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...
CAFÉ DEMOCRATICO	http://dbpedia.org/...	http://dbpedia.org/...	http://schema.org/...	http://schema.org/...

Búsqueda por nombre "DON FRANCISCO EL SABOR ORIGINAL" y su resultado

The image shows two overlapping windows from a software application. The top window, titled "Datos de búsqueda", has a search interface with a dropdown menu set to "nombre", a "SEARCH" button, and a label "Nombre:" followed by a text box containing "ABOR ORIGINAL". The bottom window, titled "Data", displays a table with the following data:

Lugar-Turístico	Actividad	Clasificación	Categoría	Local	Dirección
DON FRANCISC...	Alimentos y Bebi...	Restaurante	SEGUNDA	PRINCIPAL	Imbabura N4-54...

Búsqueda por actividad “opcional” y su resultado

Datos de búsqueda

actividad

Actividad: Alimentos y Bebidas

Lugar-Turís...	Nro. teléfono	Email	WEB	Nro. mesas	Nro. camas	Nro. habita...
EL POLLO P...	5,94E+11	ofeliallivigan...	www	18	0	0
EL SABROS...	2903118	anmevive@h...	www	13	0	0
HELADERIA ...	2568121	oficina@hela...	WWW.helad...	11	0	0
KAYA CHOC...	5,94E+11	rafaelatel...	www.kayach...	8	0	0
LA CAVA NO...	5,94E+11	-mauritziohur...	www	10	0	0
LOS EXQUIS...	5,94E+11	jvf_audioyvi...	www	15	0	0
LOS EXQUISI...	5,94E+11	jvf_audioyvi...	www	15	0	0
\RESTAURA...	5,94E+11	luislopezcard...	www	21	0	0
222 SANDUC...	5,94E+11	sebasamanie...	www	3	0	0
ARKADIUS BKS	5,94E+11	contabilidads...	www	12	0	0
BERAKÁH DE...	5,94E+11	darias04@ou...	www	13	0	0
BUON MANGI...	5,94E+11	mapu_uv@h...	www	5	0	0
BURGER KING	22222617	sbernal@bur...	www	45	0	0
C.A.C.T.U.S ...	2549591	marcoant69_...	www	7	0	0
CACTUS LA ...	5,94E+11	olgapozo12...	www	10	0	0
CAFÉ DEMO...	999022672	indiobenitez...	www	12	0	0
CANDDY DE...	5,94E+11	-ruthcazares...	www	63	0	0
CASA SAN VI...	5,94E+11	fervalencia5...	www	12	0	0
CEVICHÉRÍA ...	5,94E+11	guazcopatrici...	www	11	0	0
CHOCOLATE...	5,94E+11	alianzaquich...	www	7	0	0