Projeto 8

LCC - 3º ano

Mapeamento entre BD do Clero Catedralício da Idade Média e Ontologia CIDOC-CRM

Orientadores: Pedro Rangel Henriques (DI) e Ricardo Giuliani Martini

Nelson Sá
70925

Rafael Silva
ae4282

4 de Julho de 2018

Conteúdo

1 Introdução **4**
1.1 Definição de Modelo Relacional 4
1.2 Definição de Ontologia 5
1.3 Liguangens de queries 6
1.3.1 Structered Query Language (SQL) 6
1.3.2 SPARQL Protocol and RDF Query Language 6

2 Análise e Especificação **7**
2.1 Descrição da Base de Dados em estudo 7
2.2 Estruturas de Dados 9
2.2.1 Base de Dados 9

3 Ontologia **10**
3.1 Criação de ontologia 10
3.2 Estruturas de Dados 13
3.2.1 Protégé 13

4 Mapping **14**

5 Interrogação à ontologia via SPARQL **17**

6 Interface Web **25**

7 Conclusão **27**

A Código SPARQL das questões realizadas **29**

B Código do Mapping **33**

Lista de Figuras

1.1 Ontologia 6

2.1 Base de Dados (siepfinal) 8

3.1 Ontologia 10
3.2 Classes 11
3.3 Object Properties 12
3.4 Data Properties 13

4.1 Mapping relacionamento 15
4.2 Mapping atributo 16

5.1 SPARQL 17
5.2 Query SPARQL, relacionamento Clero têm oficio 18
5.3 Query SQL, relacionamento Clero têm oficio 18
5.4 Query SPARQL, informação de Clero 19
5.5 Query SQL, informação de Clero 19
5.6 Query SPARQL, Documento pertence ao arquivo 20
5.7 Query SQL, Documento pertence ao arquivo 20
5.8 Query SPARQL, Clero têm função 21
5.9 Query SQL, Clero têm função 21
5.10 Query SPARQL, Clero têm testamento 22
5.11 Query SQL, Clero têm testamento 22
5.12 Query SPARQL, Clero têm filho 23
5.13 Query SQL, Clero têm filho 23

5.14 Query SPARQL, Documento controlado por Historiador 24

6.1 Página Inicial da Interface Web 25

6.2 Página com os resultados da pesquisa dos respetivos ofícios de todos os clérigos 26

Capítulo 1

Introdução

Este trabalho prático enquadra-se na unidade curricular de Projeto do curso de Licenciatura em Ciências da Computação da Universidade do Minho. E o que se pretende é que os estudantes explorem um tema ou "problema" proposto pelo(s) orientador(es) e encontrem soluções adequadas.

O projeto que nos foi proposto (Projeto 8), tem como base fazer a relação do source (base de dados relacional) e o target (ontologia) através do framework Ontop). Esse projeto integra-se no trabalho de doutoramento de Ricardo Giuliani Martini.

Este documento está organizado em 7 capítulos, abordando cada etapa do desenvolvimento do projeto.

1.1 Definição de Modelo Relacional

Nesta secção, será feita uma breve introdução ao Modelo Relacional, que iremos analisar em detalhe mais à frente. O Modelo Relacional, proposto por Edgar Codd, da IBM de divulgação reduzida (Codd,1969) e, no ano seguinte, numa revista científica (Codd 1970), alterou a forma como se utilizavam as bases de dados, que consistia essencialmente na manipulação de estruturas físicas, muito próximas da representação informática em suporte físico. Ao permitir a utilização de uma linguagem de mais alto nível e abstração, independente da representação física de dados, o modelo relacional de dados contribuiu para a vulgarização desta tecnologia, melhorando a produtividade dos utilizadores das bases de dados. Assim, os utilizadores e os programadores já não tinham de se preocupar com estruturas de apontadores físicos, nem em modificar os seu programas sempre que a estrutura de dados sofriam uma alteração.

O modelo relacional de dados, permite a utilização de uma única estrutura

Nome	Sigla	ECTS
Base de Dados	BD	5
Programação	PG	4

de dados, chamada tabela que uma base de dados é uma coleção de tabelas. Observe-se a representação da tabela DISCIPLINA, em cima exemplificada.

As colunas identificam as características de uma disciplina, e cada coluna é encabeçada por um titulo. As linhas, ou tuplos, representam as diferentes disciplinas que existem na base de dados. Adicionalmente, cada coluna tem um determinado tipo de dados; por exemplo, a coluna ECTS é do tipo inteiro, a coluna Nome é do tipo texto. Uma tabela é definida pela suas colunas, cujo conjunto representa o seu esquema. Esta representação visual de uma tabela é uma forma conveniente e simples de compreender a base de dados e esconde aspetos técnico da representação informática desta estrutura. Para um utilizador não técnico, assemelha-se a uma folha de calculo; para um programador, assemelha-se a uma estrutura do tipo record(registo). (3)

1.2 Definição de Ontologia

Ontologia define-se como especificação formal e explicita de uma partilha de conceptualização, em que a conceptualização refere-se a um modelo abstrato de um fenómeno do mundo.

A ontologia vai ter um domínio (chamado classes) propriedades para cada conceito descrevendo características e atributos de um conceito. A ontologia difere-se de outros modelos de dados pela preocupação com conceitos e relacionamentos.

Uma ontologia define um comum vocabulário para os pesquisadores que têm a necessidade de partilha de informação num domínio incluindo "machine understandable"de definição de conceitos básicos.

Razões para desenvolver ontologias:
• Para partilhar entendimento comum de uma estrutura de informação entre pessoas e agentes de software.
• Para habilitar o reuso do domínio de conhecimento.
• Para fazer assumpções explicitas de domínio.
• Para separar domínio de conhecimento operacional.
• Para analisar domínio de conhecimento.

O primeiro ponto em cima, é o mais comum em desenvolver ontologias.
Suponhamos diferentes web sites utilizam o mesmo domínio de especificação. Se os web sites partilham a mesma ontologia, com os mesmos conceitos e classificações, então agentes de computadores podem por junta dos diferentes web sites.

Os agentes podem usar essa informação agregada para responder a queries. Para criarmos uma ontologia, a melhor coisa a fazer é a pesquisa de uma ontologia com o mesmo domínio, já feita, para a reutilizar.(1)

Figura 1.1: Ontologia

1.3 Liguangens de queries

1.3.1 Structered Query Language (SQL)

Para manipular as tabelas, foi definida uma linguagem, álgebra relacional que permite, com base num pequeno conjunto de operadores, construir consultas que podem ser executadas pela base de dados. Na realidade, as tabelas são manipuladas com a linguagem Structered Query Language (SQL), que implementa muitos aspetos das álgebra relacional.(3)

1.3.2 SPARQL Protocol and RDF Query Language

SPARQL (um acrónimo recursivo para SPARQL Protocol e RDF Query Language) é uma linguagem de consulta RDF, ou seja, uma linguagem de consulta semântica para base de dados, capaz de recuperar e manipular dados armazenados no Resource Description Framework (RDF). Foi feito um padrão pelo grupo de trabalho do acesso de dados RDF (DAWG) do World Wide Web Consortium, e é reconhecido como uma das tecnologias chaves da Web semântica. Em 15 janeiro 2008, SPARQL 1.0 transformou-se um oficial Recomendação W3C, e SPARQL 1.1 em março de 2013.

O SPARQL permite que uma consulta consista em padrões triplos, conjunções, disjunções e padrões opcionais.

Capítulo 2

Análise e Especificação

2.1 Descrição da Base de Dados em estudo

O projeto Fasti Ecclesiae Portugaliae tem um objetivo básico para uma base de dados relacional para o estudo do clero catedralício português na idade média. A principal ideia por detrás desse projeto é, fazer com que os ativos estejam disponíveis para a comunidade científica.
A data dos documentos, incluindo a base de dados (bdFasti) é entre 1071, ano da restauração da diocese de Braga depois da ocupação muçulmana, 1325, fim do reinado do D.Dinis.
O projeto engloba mais de 700 documentos de 9 dioceses(Braga, Porto, Coimbra, Lamego, Guarda, Viseu, Lisboa, Évora e Silves).
O estudo sobre o Clero é baseado em análise de bispos e arcebispos, também dignidades, cânones, constituição de porções de catedrais, excluindo o resto do pessoal auxiliar do clero como capelães, vigários entre outros.
Toda a informação está armazenada na base de dados "siepfinal".(1)

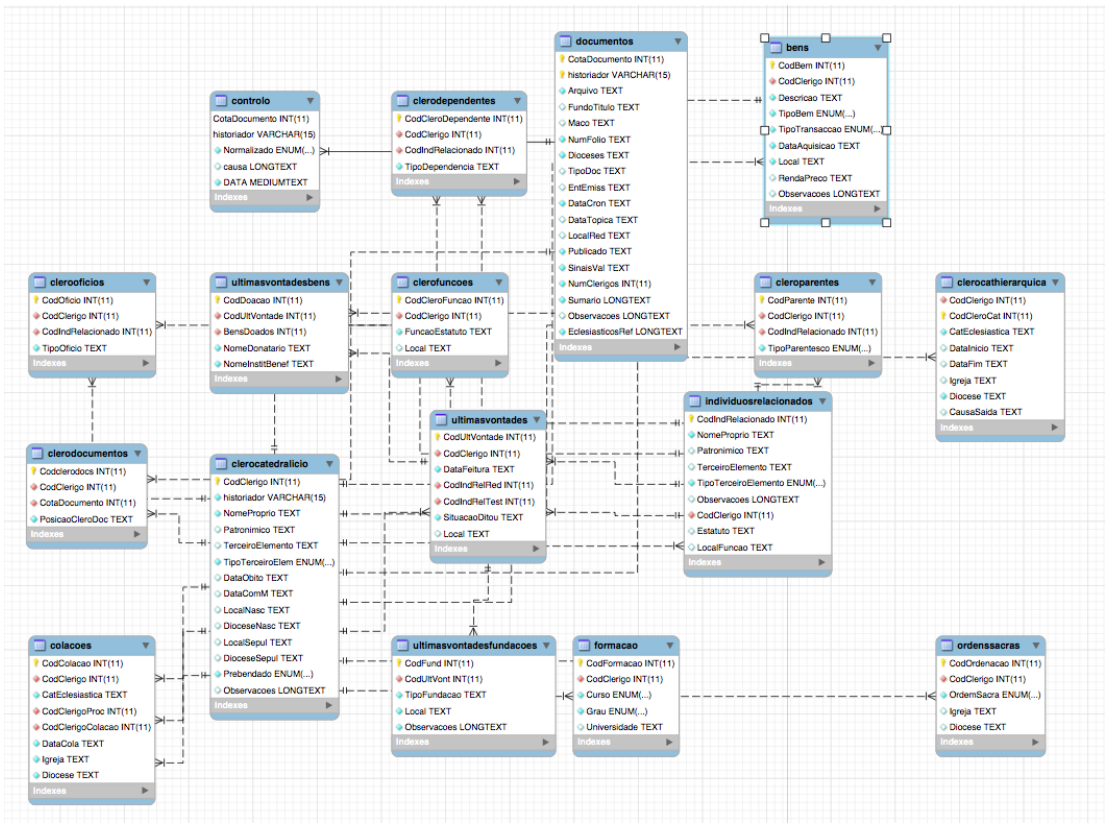

Figura 2.1: Base de Dados (siepfinal)

2.2 Estruturas de Dados

2.2.1 Base de Dados

INT CodOficio, Codclerodocs, CodColacoes, CodDoacao, CodCleroFuncao, CodCleroDependente, CodUltVontade, CotaDocumento, CodFund, CodFormacao, CodParente, CodIndRelacionado, CodCleroCat, CodOrdenacao, CodBem, NumClerigos;

VARCHAR historiador;

ENUM Normalizado, TipoTerceiroElem, Prebendado, Curso, Grau, TipoTransaccao, TipoParentesco, TipoTerceiroElemnto, OrdemSacra;

TEXT TipoOficio, PosicaoCleroDoc, CatEclesiatica, DataCola, Igreja, Diocese, DataMedium, NomeDonatario, NomeInstiBenef, NomeProprio, Patronimico, TerceiroElemento, DataObito, DataComM, LocalNasc, DioceseNasc, LocalSepul, DioceseSepul, TipoDependencia, FuncaoEstatuto, Local, DataFeitura, TipoFundacao, Arquivo, FundoTitulo, Maco, NumFolio, Dioceses, TipoDoc, EntEmiss, DataCron, DataTopica, LocalRed, Publicado, SinaisVal, Universisade, Descricao, DataAquisacao, RendaPreco, Estatuto, LocalFuncao, DataInicio, DataFim, causaSaida;

LONGTEXT causa, Observacoes, Sumario, EclesiaticosRef;

Capítulo 3

Ontologia

3.1 Criação de ontologia

Este projeto, como referido em cima, tem como objetivo criar uma ontologia de maneira a simplificar a leitura de uma base de dados relacional. Assim, o primeiro passo é desenhar um modelo conceptual da ontologia, partindo de uma base de dados relacional. Utilizando a ferramenta draw.io foi obtido o resultado apresentado na figura 3.1. Este modelo foi construido a partir da base de dados em estudo, "siepfinal", Figura 2.1.

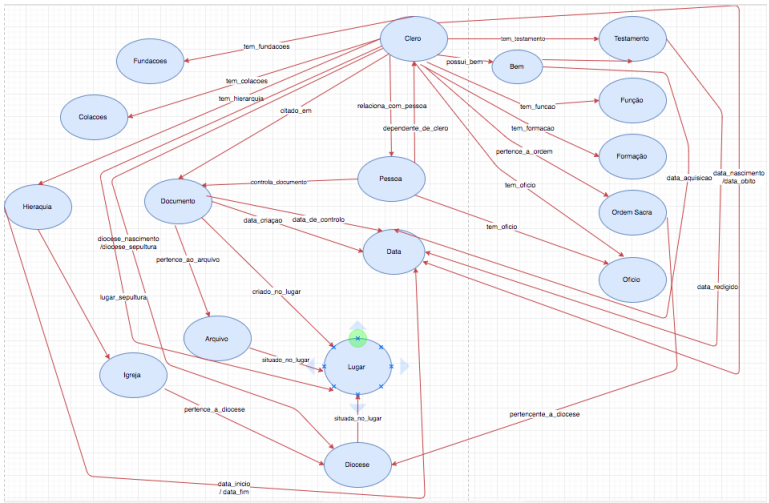

Figura 3.1: Ontologia

A partir do modelo da Figura 3.1, torna-se mais fácil implementar a ontologia em questão, utilizando a ferramenta protégé. Para a criação de uma ontologia, no protégé, são criadas classes correspondentes à da figura 3.1 (por exemplo, para a tabela "clerocatedralicio" da Figura 2.1 é criada uma classe Clero, no protégé) como vemos na figura em baixo.

Alguns atributos da base de dados relacional podem ser transformados em classes, na ontologia, de forma a facilitar a sua consulta (por exemplo, o atributo Arquivo da tabela "documentos" da base de dados relacional torna-se a classe Arquivo, na ontologia. Por consequente, sabe-se que toda a informação dos arquivos vão estar na classe Arquivos).

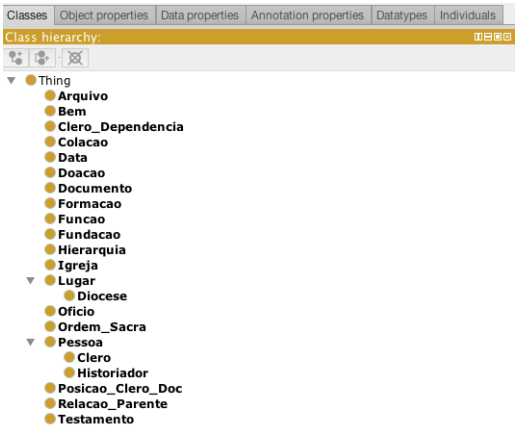

Figura 3.2: Classes

São ainda criados relacionamentos entre as classes ("object properties", no protégé), em que cada uma tem um nome, como pode ser visto na figura em baixo. (por exemplo, a classe Clero é citada num Documento, portanto tem-se a relação: Clero (Domains) citado_em Documento (Ranges).

Figura 3.3: Object Properties

E, por último, são criados os atributos (“data properties“, no protégé), atributos esses que estão na base de dados relacional.

Figura 3.4: Data Properties

3.2 Estruturas de Dados

3.2.1 Protégé

string catEclesiastica, causa, causaSaida, dataTopica, decricao, eclesiasticosRef, entEmiss, estatuto, fundoTitulo, maco, numFolio, observacoes, patrimonio, publicado, rendaPreco, sinaisVal, situacaoDitou, sumario, terceiroElemento, tipoDependencia, tipoDoc, tipoOficio, universidade, data_de_aquisacao, data_de_controlo, data_de_criacao, data_de_nascimento, data_de_obito, data_fim, data_inicio;

range curso, grau, normalizado, ordemSacra, prebendado, tipoBem, tipoParentesco, tipoTerceiroElemento, tipoTransacao;

int numClerigos;

Name nomeProprio;

Capítulo 4

Mapping

O mapping, sucintamente, é um mecanismo de ponte entre a base de dados relacional e a ontologia OWL, ou seja. Então, o objetivo é criar queries na camada de dados através da a camada conceptual.
Ontology-Base Data Acess (OBDA) é um paradigma que providencia um acesso semântico entre base de dados por meios de ontologias. Assim sendo o objetivo do OBDA é acessar dados através da ontologia.
No OBDA, existe uma camada abstrata como uma ontologia que define um vocabulário partilhado de um domínio especifico. OBDA esconde a estrutura de base de dados e com isso pode enriquecer dados incompletos com conhecimento prévio. OBDA é baseado em arquitetura terceiro nível estabelecido por uma ontologia (componente principal e uma descrição formal do domínio de interesse), dados fontes e os mapeamentos que ligam os dois primeiros.
O ONTOP mapping necessita de conectar as classes e relações (Classe e object properties) de uma ontologia com views (SQL) através da base de dados.(1)

- mappingId: string que define axioma de mapeamento
- target: um modelo triplo de ontologia (sujeito, predicado, objeto) que faz referência a nomes de colunas usado na base de dados(source)
- source: query em SQL implementada na base de dados relacional

Como vimos no capitulo 3, já foi criado as Classes e as relações (object properties) entre as Classes, vamos então fazer o mapping entre a ontologia e base de dados, para isso fizemos em duas fazes:

1º Fase

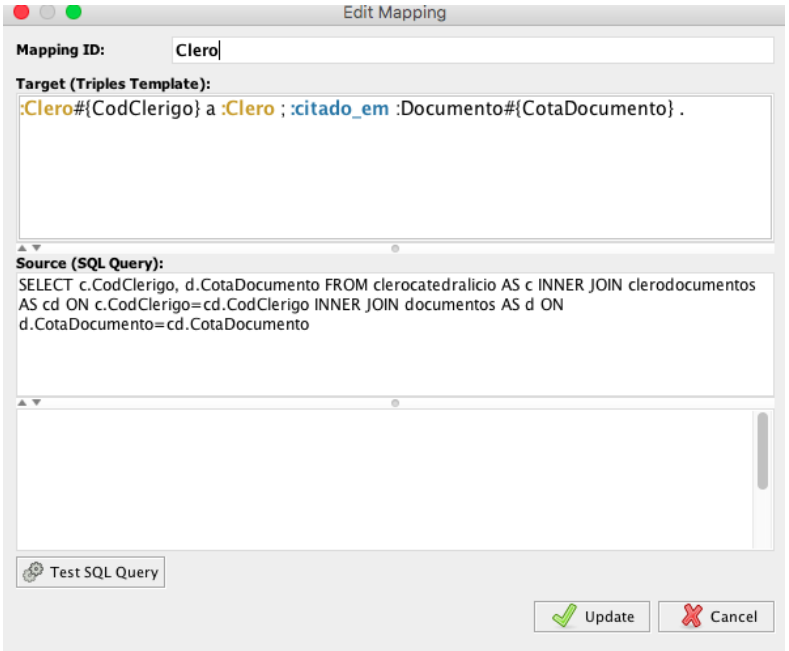

Figura 4.1: Mapping relacionamento

No Target(Triples Template), colocamos o triplo como vimos em cima, em que a Classe Clero com o CodClerigo como identificador, do tipo Clero tem um relacionamento(object propertie) com a Classe Documento com identificador CotaDocumento em que o relacionamento se chama citado_em.

No Source(Query SQL) colocamos a query em SQL que vai ligar a base de dados aos triplos.

2º Fase

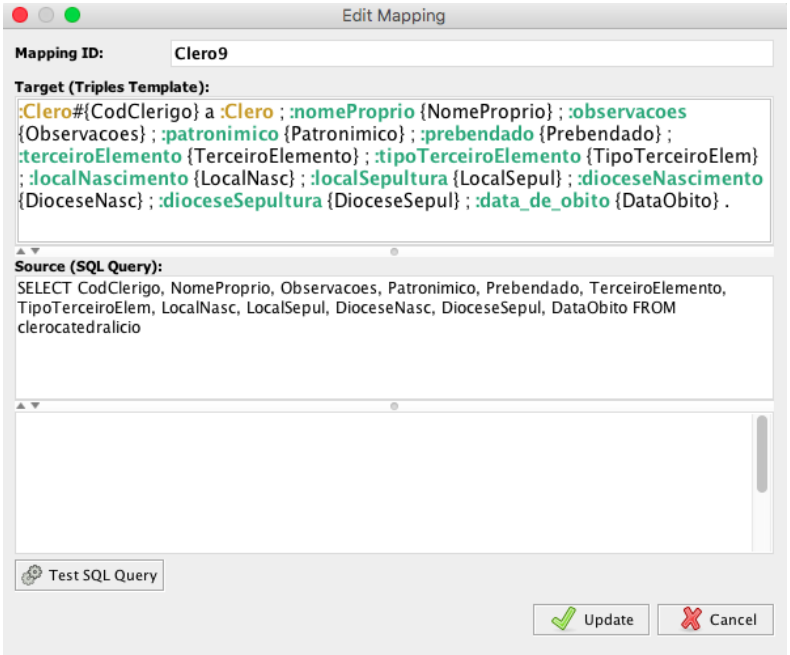

Figura 4.2: Mapping atributo

No Target(Triples Template), colocamos o triplo como vimos em cima, em que a Classe Clero com o CodClerigo como identificador, do tipo Clero tem vários atributos(data properties) como por exemplo nomeProprio que vai ligar ao atributo da base de dados chamado NomeProprio.

No Source(Query SQL) colocamos a query em SQL que vai ligar a base de dados aos triplos.

Fazendo isso para todos os relacionamentos e atributos no caso em estudo, para mais detalhes do mapping pode consultar o Apêndice B.

Capítulo 5

Interrogação à ontologia via SPARQL

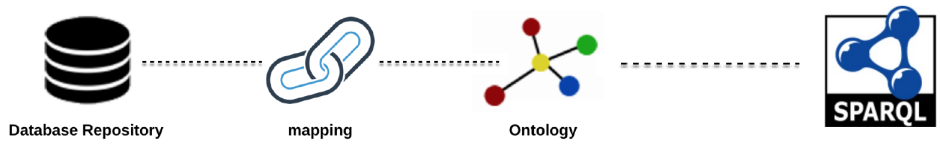

Figura 5.1: SPARQL

Vamos interrogar a ontologia via query SPARQL, sendo que o resultado têm que ser igual ao de interrogar diretamente à base de dados, como vamos ver de seguida.

ontop query editor:

Query Editor

```
PREFIX : <http://www.semanticweb.org/nelsonsa12/ontologies/2018/3/untitled-ontology-12#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX quest: <http://obda.org/quest#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?Clero ?nome ?tipoOf
WHERE {
  ?Clero :tem_oficio ?Oficio.
  ?Clero :nomeProprio ?nome.
  ?Oficio :tipoOficio ?tipoOf.

}
```

Execution time: 0.035 sec – Number of rows retrieved: 1,028 | Show: 0 | All | Short IRI | Attach Prefixes | Execute | Save Changes

Clero	nome	tipoOf
:Clero#10	"Johane"	"Ouvidor do arcebispo"^^xsd:string
:Clero#10	"Johane"	"Ouvidor do arcebispo"^^xsd:string
:Clero#83	"D. Johanne"	"Ouvidor do arcebispo"^^xsd:string
:Clero#91	"D. Johannis"	"Procurador do arcebispo"^^xsd:string
:Clero#121	"Tello (D. Frei)"	"procurador do arcebispo em Terra de Mateus"^^xsd:str...
:Clero#121	"Tello (D. Frei)"	"Procurador do arcebispo"^^xsd:string
:Clero#156	"Frater S. "	"Procurador "^^xsd:string
:Clero#157	"Johannis "	"Vigário do arcebispo"^^xsd:string
:Clero#160	"Martinus, "	"Procurador "^^xsd:string
:Clero#160	"Martinus, "	"Procurador "^^xsd:string
:Clero#170	"Frater Stephanus"	"Vigário."^^xsd:string
:Clero#170	"Frater Stephanus"	"Vigário."^^xsd:string
:Clero#170	"Frater Stephanus"	"vigário do bispo."^^xsd:string
:Clero#181	"Gunsaluus"	"Procurador "^^xsd:string
:Clero#191	"Martinho"	"mordomo e procurador"^^xsd:string
:Clero#203	"Gunsaluus"	"Advogado"^^xsd:string
:Clero#203	"Gunsaluus"	"Advogado"^^xsd:string
:Clero#205	"D. Martinho"	"procurador "^^xsd:string
:Clero#241	"Martinus"	"Mordomo"^^xsd:string

Figura 5.2: Query SPARQL, relacionamento Clero têm oficio

```
use siepfinal;

select c.CodClerigo, c.NomeProprio, o.TipoOficio
                from clerocatedralicio as c
                    inner join clerooficios as o
                            on c.CodClerigo = o.CodClerigo
```

100% 1:8

Result Grid | Filter Rows: Search | Export:

CodClerigo	NomeProprio	TipoOficio
10	Johane	Ouvidor do arcebispo
10	Johane	Ouvidor do arcebispo
83	D. Johanne	Ouvidor do arcebispo
91	D. Johannis	Procurador do arcebispo
121	Tello (D. Frei)	procurador do arcebispo em Terra de Mateus
121	Tello (D. Frei)	Procurador do arcebispo
156	Frater S.	Procurador
157	Johannis	Vigário do arcebispo
160	Martinus,	Procurador
160	Martinus,	Procurador
170	Frater Stephanus	Vigário.
170	Frater Stephanus	Vigário.
170	Frater Stephanus	vigário do bispo.
181	Gunsaluus	Procurador
191	Martinho	mordomo e procurador
203	Gunsaluus	Advogado
203	Gunsaluus	Advogado
205	D. Martinho	procurador
241	Martinus	Mordomo

Figura 5.3: Query SQL, relacionamento Clero têm oficio

Na Figura 5.2 e 5.3 vimos uma query em SPARQL e SQL, em que a Classe Clero têm um relacionamento com a Classe Oficio com o nome tem_oficio, para melhor compreender fizemos select do identificador do Clero, do nome e o

respetivo oficio.

ontop query editor:

Query Editor

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX quest: <http://obda.org/quest#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?Clero ?nome ?patroni ?terceiroE ?tipoTerceiroE ?dataO ?localNasc ?dioceseNasc ?localSep ?dioceseSep ?prebend ?obser
WHERE {
  ?Clero :nomeProprio ?nome.
  ?Clero :patronimico ?patroni.
  ?Clero :terceiroElemento ?terceiroE.
  ?Clero :tipoTerceiroElemento ?tipoTerceiroE.
  ?Clero :data_de_obito ?dataO.
  ?Clero :localNascimento ?localNasc.
  ?Clero :dioceseNascimento ?dioceseNasc.
  ?Clero :localSepultura ?localSep.
  ?Clero :dioceseSepultura ?dioceseSep.
  ?Clero :prebendado ?prebend.
  ?Clero :observacoes ?obser.
}
```

Execution time: 0.187 sec - Number of rows retrieved: 19,267 Show: 0 All Short IRI Attach Prefixes Execute Save Changes

Clero	nome	patroni	terceiroE	tipoTerceiroE	dataO	localNasc	dioceseNasc	localSep	dioceseSep	prebend	obser
:Clero#1	"Martinus, "	"Iohannis"^^...	"Barrosas"^...	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#2	"Joham"	"Siluestre"^^...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#3	"Dominicus"	"Dominicj"^...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#4	"Dominicus"	"Rial"^^xsd:...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#5	"Bartholome...	""^^xsd:string	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#6	"Stephanus"	"Iohannis"^^...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#7	"Valascus"	"Martinj"^^x...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#8	"Stephanus"	"Pelagij"^^x...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#9	"Johannes"	"Siluestri"^^...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#10	"Johane"	""^^xsd:string	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#11	"Domingos"	"Dominguiz"...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#12	"Domingos "	"Perez"^^xs...	""^^xsd:string	"Ausente"	"Já falecido ...	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#13	"Domingos "	"Perez"^^xs...	""^^xsd:string	"Ausente"	"Já falecido ...	""	""	""	""	"seminforma...	"Instituiu um...
:Clero#14	"Joham"	"Siluestris"^...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#15	"Joham"	"Martijnz"^^...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#16	"Duram"	"Steuaez"^^...	"dito de Rial...	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#17	"Steue"	"Annes"^^xs...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#18	"Joham"	"Annes"^^xs...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	""^^xsd:string
:Clero#19	"Joham"	"Annes"^^xs...	""^^xsd:string	"Ausente"	""^^xsd:string	""	""	""	""	"seminforma...	"Herdeiro d...

Figura 5.4: Query SPARQL, informação de Clero

```
1  use siepfinal;
2
3  select CodClerigo, NomeProprio, Patronimico, TerceiroElemento TipoTerceiroElem, DataObito, LocalNasc,
4         DioceseNasc, LocalSepul, DioceseSepul, Prebendado, Observacoes
5            from clerocatedralicio
```

100% 33:5

Result Grid Filter Rows: Search Edit: Export/Import: Fetch rows:

CodClerigo	NomeProprio	Patronimico	TipoTerceiroElem	DataObito	LocalNasc	DioceseNasc	LocalSepul	DioceseSepul	Prebendado	Observacoes
1	Martinus,	Iohannis	Barrosas						seminformação	
2	Joham	Siluestre							seminformação	
3	Dominicus	Dominicj							seminformação	
4	Dominicus	Rial							seminformação	
5	Bartholomeus								seminformação	
6	Stephanus	Iohannis							seminformação	
7	Valascus	Martinj							seminformação	
8	Stephanus	Pelagij							seminformação	
9	Johannes	Siluestri							seminformação	
10	Johane								seminformação	
11	Domingos	Dominguiz							seminformação	
12	Domingos	Perez		Já falecid...					seminformação	
13	Domingos	Perez		Já falecid...					seminformação	Instituiu um...
14	Joham	Siluestris							seminformação	
15	Joham	Martijnz							seminformação	
16	Duram	Steuaez	dito de Rial						seminformação	
17	Steue	Annes							seminformação	
18	Joham	Annes							seminformação	
19	Joham	Annes							seminformação	Herdeiro do...

Figura 5.5: Query SQL, informação de Clero

Na Figura 5.4 e 5.5 vimos uma query em SPARQL e SQL, em que fazemos um

select de toda informação da Classe Clero.

ontop query editor:

Query Editor

```
PREFIX : <http://www.semanticweb.org/nelsonsa12/ontologies/2018/3/untitled-ontology-12#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX quest: <http://obda.org/quest#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?Arquivo
WHERE {
    ?Documento :pertence_ao_arquivo ?Arquivo.
}
```

Execution time: 0.164 sec - Number of rows retrieved: 15 | Show: 0 | All | Short IRI | Attach Prefixes | Execute | Save Changes

```
Arquivo
:Arquivo#ADB%20-%20Arquivo%20Distrital%20de%20Braga
:Arquivo#TT%20-%20Instituto%20dos %20Arquivos%20Nacionais%2FTorre%20do%20Tombo
:Arquivo#ADP%20-%20Arquivo%20Distrital%20do%20Porto
:Arquivo#TT%20-%20Instituto%20dos%20Arquivos%20Nacionais%2FTorre%20do%20Tombo
:Arquivo#ACPL-%20Arquivo%20da%20Cúria%20Patriarcal%20de%20Lisboa
:Arquivo#TT%20-%20Instituto%20dos%3F%20Arquivos%20Nacionais%2FTorre%20do%20Tombo
:Arquivo#BN%20-%20Biblioteca%20Nacional
:Arquivo#AUC%20-%20Arquivo%20da%20Universidade%20de%20Coimbra
:Arquivo#ADE%20-%20Arquivo%20Distrital%20de%20Évora
:Arquivo#AMG%20-%20Arquivo%20Municipal%20%28Alfredo%20Pimenta%29%20de%20Guimarães
:Arquivo#ACE-Arquivo%20do%20Cabido%20da%20Sé%20de%20Évora
:Arquivo#ACZamora%20-%20Arquivo%20da%20Catedral%20de%20Zamora
:Arquivo#AML%20-%20Arquivo%20Municipal%20de%20Lisboa
:Arquivo#ADV%20-%20Arquivo%20Distrital %20de%20Viseu
:Arquivo#desconhecido
```

Figura 5.6: Query SPARQL, Documento pertence ao arquivo

```
1  use siepfinal;
2
3  select distinct arquivo from documentos
4
5
6
```

100% 1:6

Result Grid | Filter Rows: Search | Export:

arquivo
ADB - Arquivo Distrital de Braga
TT - Instituto dos Arquivos Nacionais/Torre do T...
ADP - Arquivo Distrital do Porto
TT - Instituto dos Arquivos Nacionais/Torre do T...
ACPL- Arquivo da Cúria Patriarcal de Lisboa
TT - Instituto dos? Arquivos Nacionais/Torre do...
BN - Biblioteca Nacional
AUC - Arquivo da Universidade de Coimbra
ADE - Arquivo Distrital de Évora
AMG - Arquivo Municipal (Alfredo Pimenta) de...
ACE-Arquivo do Cabido da Sé de Évora
ACZamora - Arquivo da Catedral de Zamora
AML - Arquivo Municipal de Lisboa
ADV - Arquivo Distrital de Viseu
desconhecido

Figura 5.7: Query SQL, Documento pertence ao arquivo

Na Figura 5.6 e 5.7 vimos uma query em SPARQL e SQL, em que dá informação

da localização(Arquivo) da Classe Documento.

ontop query editor:

Query Editor

```
PREFIX : <http://www.semanticweb.org/nelsonsa12/ontologies/2018/3/untitled-ontology-12#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX quest: <http://obda.org/quest#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?nomeProprio ?funcao ?locate
WHERE {
    ?clero :tem_funcao ?func .
    ?clero :nomeProprio ?nomeProprio .
    ?func :funcaoEstatuto ?funcao .
    ?func :local ?locate .
    FILTER(?nomeProprio != "desconhecido" && regex(?locate,"Porto","i"))

}
```

Execution time: 0.106 sec - Number of rows retrieved: 14 Show: 0 All Short IRI Attach Prefixes Execute Save Changes

nomeProprio	funcao	locate
"Petrus"	"Juiz"^^xsd:string	"Porto"
"Petrus"	"juiz "^^xsd:string	"Cidade do Porto"
"Dominicus"	"vigário do bispo."^^xsd:string	"Porto"
"Gundisaluus"	"vigário do bispo."^^xsd:string	"Porto"
"Gillelmi"	"vigário do bispo."^^xsd:string	"Porto"
"Johannes"	"Dom"^^xsd:string	"Cabido do Porto"
"Gonsalvus"	"Dom"^^xsd:string	"Cabido de Coimbra/Porto"
"Martinus"	"Vigário"^^xsd:string	"Porto"
"Johannes"	"Procurador do cabido de Viseu."^^xsd:string	"Porto"
"Martinus"	"Procurador do cabido da Guarda"^^xsd:string	"Porto"
"Johannes"	"Procurador do cabido de Viseu."^^xsd:string	"Porto"
"Martinus"	"Procurador do cabido da Guarda"^^xsd:string	"Porto"
"Johannes"	"Procurador do cabido de Viseu."^^xsd:string	"Porto"
"Martinus"	"Procurador do cabido da Guarda"^^xsd:string	"Porto"

Figura 5.8: Query SPARQL, Clero têm função

SQL File 4* SQL File 3*

```
use siepfinal;

select c.NomeProprio, f.FuncaoEstatuto, f.Local
                from clerocatedralicio as c
                    inner join clerofuncoes as f
                            on c.CodClerigo = f.CodClerigo where c.NomeProprio not like "desconhecido" and f.Local like "%Porto"
```

Result Grid Filter Rows: Export:

NomeProprio	FuncaoEstatuto	Local
Petrus	Juiz	Porto
Petrus	juiz	Cidade do Porto
Dominicus	vigário do bispo.	Porto
Gundisaluus	vigário do bispo.	Porto
Gillelmi	vigário do bispo.	Porto
Johannes	Dom	Cabido do Porto
Gonsalvus	Dom	Cabido de Coimbra/Porto
Martinus	Vigário	Porto
Johannes	Procurador do cabido de Viseu.	Porto
Martinus	Procurador do cabido da Guarda	Porto
Johannes	Procurador do cabido de Viseu.	Porto
Martinus	Procurador do cabido da Guarda	Porto
Johannes	Procurador do cabido de Viseu.	Porto
Martinus	Procurador do cabido da Guarda	Porto

Figura 5.9: Query SQL, Clero têm função

Na Figura 5.8 e 5.9 vimos uma query em SPARQL e SQL, em que dá nome do Clero a sua função que desempenha e localidade em que exerce.

ontop query editor:

Query Editor

```
SELECT ?name ?testamento ?doacao ?bem ?desc
WHERE {
{
  ?clero :tem_testamento ?testamento.
  ?clero :nomeProprio ?name

}

 {
  ?testamento :envolve_doacao ?doacao
  }

 {
  ?doacao :envolve_bem ?bem .
  ?bem :descricao ?desc
  }
}
```

Execution time: 0.018 sec - Number of rows retrieved: 3 Show: 0 All Short IRI Attach Prefixes Execute Save Changes

name	testamento	doacao	bem	desc
"A. [Arias]"	:Testamento#1	:Doacao#1	:Bem#75	"Uma Biblia"^^xsd:string
"Vincentius"	:Testamento#4	:Doacao#2	:Bem#1028	"Uma herdade "^^xsd:string
"Petrus (Magister)"	:Testamento#6	:Doacao#3	:Bem#1864	"inexistente"^^xsd:string

Figura 5.10: Query SPARQL, Clero têm testamento

SQL File 4* SQL File 3* Don't Limit

```
use siepfinal;


SELECT c.NomeProprio, t.CodUltVontade, d.CodDoacao, b.CodBem, b.Descricao
       FROM clerocatedralicio as c
           INNER JOIN ultimasvontades AS t
             ON c.CodClerigo = t.CodClerigo
              INNER JOIN ultimasvontadesbens AS d
                  ON t.CodUltVontade=d.CodUltVontade
                     INNER JOIN bens AS b
                         ON d.BensDoados=b.CodBem
```

100% 1:13

Result Grid Filter Rows: Search Export:

NomeProprio	CodUltVontade	CodDoacao	CodBem	Descricao
A. [Arias]	1	1	75	Uma Biblia
Vincentius	4	2	1028	Uma herdade
Petrus (Magister)	6	3	1864	inexistente

Figura 5.11: Query SQL, Clero têm testamento

Na Figura 5.10 e 5.11 vimos uma query em SPARQL e SQL, que dá o nome do Clero, o seu testamento, o identificador da Doação e Bem, e respetiva descrição do Bem.

ontop query editor:

Query Editor

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?nomeproprioC ?tipoparentesco ?nomeproprioP
WHERE {
    {
    ?clero :tem_relacao_parente ?relacao.
    ?clero :nomeProprio ?nomeproprioC.
    ?relacao :tipoParentesco ?tipoparentesco.
    FILTER(?tipoparentesco = "Filho").

    }
    {
     ?relacao :e_pessoa_parente ?pessoa.
     ?pessoa :nomeProprio ?nomeproprioP.
     FILTER(?nomeproprioP != "desconhecido").
    }

}
```

Execution time: 0.019 sec - Number of rows retrieved: 9 | Show: 0 | All | Short IRI | Attach Prefixes | Execute | Save Changes

nomeproprioC	tipoparentesco	nomeproprioP
"Johannes"	"Filho"	"Vasco"
"Petrus"	"Filho"	"Domingos"
"Petrus"	"Filho"	"Estevão"
"Laurencius"	"Filho"	"Vasco"
"Petrus"	"Filho"	"Martim"
"Bartholomeus"	"Filho"	"Petrus"
"Johannes"	"Filho"	"Martinus"
"Laurencius (Lourenço)"	"Filho"	"Johannes (João)"
"Laurencius (Lourenço)"	"Filho"	"Egeas (Egas)"

Figura 5.12: Query SPARQL, Clero têm filho

SQL File 4* | SQL File 3*

Don't Limit

```
use siepfinal;


SELECT DISTINCT  c.NomeProprio AS Clero, p.NomeProprio AS Pessoa
        FROM clerocatedralicio AS c
            INNER JOIN cleroparentes AS cp
                ON c.CodClerigo = cp.CodClerigo
                    INNER JOIN individuosrelacionados AS p
                        ON cp.CodIndRelacionado = p.CodIndRelacionado
                            WHERE cp.TipoParentesco like "Filho" AND p.nomeProprio not like "desconhecido"
```

100% | 1:12

Result Grid | Filter Rows: Search | Export:

Clero	Pessoa
Johannes	Vasco
Petrus	Domingos
Petrus	Estevão
Laurencius	Vasco
Petrus	Martim
Bartholomeus	Petrus
Johannes	Martinus
Laurencius (Lourenço)	Johannes (João)
Laurencius (Lourenço)	Egeas (Egas)

Figura 5.13: Query SQL, Clero têm filho

Na Figura 5.12 e 5.13 vimos uma query em SPARQL e SQL, que dá o nome do Clero e o da Pessoa que tem como relação de parentesco “filho”.

ontop query editor:

Query Editor

```
PREFIX : <http://www.semanticweb.org/nelsonsa12/ontologies/2018/3/untitled-ontology-12#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX quest: <http://obda.org/quest#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT *
WHERE
{
 ?s ?p ?o .
FILTER (regex(?o, "justiniana","i"))
}
```

Execution time: 1.71 sec - Number of rows retrieved: 463 | Show: 0 | All | Short IRI | Attach Prefixes | Execute | Save Changes

s	p	o
:Documento#1	:e_controlado	:Historiador#justiniana
:Documento#2	:e_controlado	:Historiador#justiniana
:Documento#3	:e_controlado	:Historiador#justiniana
:Documento#4	:e_controlado	:Historiador#justiniana
:Documento#5	:e_controlado	:Historiador#justiniana
:Documento#6	:e_controlado	:Historiador#justiniana
:Documento#7	:e_controlado	:Historiador#justiniana
:Documento#8	:e_controlado	:Historiador#justiniana
:Documento#9	:e_controlado	:Historiador#justiniana
:Documento#10	:e_controlado	:Historiador#justiniana
:Documento#11	:e_controlado	:Historiador#justiniana
:Documento#12	:e_controlado	:Historiador#justiniana
:Documento#13	:e_controlado	:Historiador#justiniana
:Documento#14	:e_controlado	:Historiador#justiniana
:Documento#15	:e_controlado	:Historiador#justiniana
:Documento#16	:e_controlado	:Historiador#justiniana
:Documento#17	:e_controlado	:Historiador#justiniana
:Documento#18	:e_controlado	:Historiador#justiniana
:Documento#19	:e_controlado	:Historiador#justiniana

Figura 5.14: Query SPARQL, Documento controlado por Historiador

Na Figura 5.14 vimos uma query em SPARQL, que dá os documentos que são controlados pela historiadora "Justiniana".

Para ver o código SPARQL de todas as queries usadas consulte o Apêndice A

Capítulo 6

Interface Web

Esta etapa do projeto abordará a futura implementação de uma interface que facilite a interação com o utilizador. Desta forma, a interface deverá ser uma página web simples que permita ao utilizador escolher as informações a que aceder e mostrar. A pagina começará por mostrar uma tabela com toda a informação relacionada com todos os cleros.

Clero Catedralicio Portugues

Nome Clero: Relacao: Atributo:

Submit

CodClerigo	historiador	NomeProprio	Patronimico	TerceiroElemento	TipoTerceiroElem	DataObito	DataComM	LocalNasc	DioceseNasc	LocalSepul	DioceseSepul	Prebendado	Observacoes
1	justiniana	Martinus,	Iohannis	Barrosas	Ausente							seminformaÃ§Ã£o	
2	justiniana	Joham	Siluestre		Ausente							seminformaÃ§Ã£o	
3	justiniana	Dominicus	Dominicj		Ausente							seminformaÃ§Ã£o	
4	justiniana	Dominicus	Rial		Ausente							seminformaÃ§Ã£o	
5	justiniana	Barthlomeus			Ausente							seminformaÃ§Ã£o	
6	justiniana	Stephanus	Iohannis		Ausente							seminformaÃ§Ã£o	
7	justiniana	Valascus	Martinj		Ausente							seminformaÃ§Ã£o	
8	justiniana	Stephanus	Pelagij		Ausente							seminformaÃ§Ã£o	
9	justiniana	Johannes	Siluestri		Ausente							seminformaÃ§Ã£o	
10	justiniana	Johane			Ausente							seminformaÃ§Ã£o	
11	justiniana	Domingos	Dominguiz		Ausente							seminformaÃ§Ã£o	
12	justiniana	Domingos	Perez		Ausente	JÃ¡ falecido nesta data.						seminformaÃ§Ã£o	
13	justiniana	Domingos	Perez		Ausente	JÃ¡ falecido nesta data.						seminformaÃ§Ã£o	Instituiu uma capela na SÃ©, com rendimentos da sua vinha dos ChÃ£os.
14	justiniana	Joham	Siluestris		Ausente							seminformaÃ§Ã£o	
15	justiniana	Joham	Martijnz		Ausente							seminformaÃ§Ã£o	
16	justiniana	Duram	Steuaez	dito de Rial	Ausente							seminformaÃ§Ã£o	
17	justiniana	Steue	Annes		Ausente							seminformaÃ§Ã£o	

Figura 6.1: Página Inicial da Interface Web

O utilizador poderá então indicar o clero do qual quer informação (inserindo o respetivo nome no campo "Nome Clero"), indicar o tipo da relação, por exemplo "tem_oficio", (inserindo o respetivo nome no campo "Relacao") ou o atributo que pretende ver, por exemplo "tipoOficio". Depois, os dados inseridos pelo utilizador, serão usados para gerar de forma automática a querie SPARQL que será usada para extrair a informação da base de dados.

Clero Catedralicio Portugues

Nome Clero: Relacao: tem_ofício Atributo: tipoOfício

Submit

CodClerigo	NomeProprio	tipoOfício
10	Johane	Ouvidor do arcebispo
10	Johane	Ouvidor do arcebispo
83	D. Johanne	Ouvidor do arcebispo
91	D. Johannis	Procurador do arcebispo
121	Telio (D. Frei)	procurador do arcebispo em Terra de Malaus
121	Telio (D. Frei)	Procurador do arcebispo
156	Frater S.	Procurador
157	Johannis	VigÃ¡rio do arcebispo
160	Martinus,	Procurador
160	Martinus,	Procurador
170	Frater Stephanus	VigÃ¡rio.
170	Frater Stephanus	VigÃ¡rio.
170	Frater Stephanus	vigÃrio do bispo.
181	Gunsalvus	Procurador
191	Martinho	monteiro e procurador
203	Gunsalvus	Advogado
203	Gunsalvus	Advogado
205	D. Martinho	procurador
241	Martinus	Monteiro
254	Martinho	Monteiro

Figura 6.2: Página com os resultados da pesquisa dos respetivos ofícios de todos os clérigos

Com esta interface web tem-se a criação de queries em linguagem natural e exibição dos resultados em tabelas, na mesma página, facilitando a interação com utilizadores desconhecedores das ferramentas.

Capítulo 7

Conclusão

Com este projeto, era pretendida a criação de um ontologia e respetivo mapeamento sobre uma base de dados já existente.

Inicialmente, foram apresentadas as ferramentas e conceitos que iriam ser necessários para a realização do projeto e, posteriormente, foi feito o seu devido estudo.

Depois, partiu-se para a criação do diagrama da ontologia de maneira a facilitar o seu entendimento e a etapa seguinte: a criação da ontologia com a ferramenta Protégé. Estas etapas foram feitas com relativo cuidado de maneira a que as estruturas da ontologia estivessem em conformidade com a base de dados relacional.

Na etapa seguinte, procedeu-se ao mapeamento da ontologia, com o Protégé.

Na etapa final, procedeu-se à interrogação da ontologia através de queries SPARQL, utilizando o Protégé, de modo a verificar que os resultados obtidos através da ontologia são condizentes com os resultados obtidos diretamente da base de dados. Procedeu-se ainda ao inicio do desenvolvimento de uma interface web que tornasse mais amigável e natural essa interrogação à base de dados e respetiva exibição de resultados.

Assim, achamos que o projeto foi realizado de forma satisfatória podendo continuar-se o seu desenvolvimento principalmente ao nível da interface web e interrogação à base de dados.

Bibliografia

[1] *Formal Description and Automatic Generation of Learning Spaces based on Ontologies*. Ricardo Giuliani Martini

[2] *Bob DuCharme*. Learning SPARQL

[3] *Fundamentos de Bases de Dados*. Feliz Gouveia

Apêndice A

Código SPARQL das questões realizadas

```
SELECT*
WHERE {
    ?Clero :nomeProprio ?name .
    ?Clero :data_de_obito ?dataObito .
}


SELECT*
WHERE {
    ?Clero :nomeProprio ?name .
    ?Clero :dioceseNascimento ?dioceseNascimento .
}



SELECT *
WHERE {
     ?s ?p ?o .
     FILTER(regex(?o,"justiniana","i"))

}

SELECT *
WHERE {
     ?s ?p ?o .
     FILTER(regex(?s,"justiniana","i"))

}
```

```
SELECT *
WHERE {
     ?s :controla ?o .
     FILTER(regex(?s,"justiniana","i"))

}


SELECT ?Clero ?nome
{
  ?Clero :nomeProprio "Joham" .
  ?Clero :nomeProprio ?nome
}


SELECT DISTINCT ?nomeProprio ?tipoOficio
WHERE {
       ?nome :tem_oficio ?tipoO .
       ?nome :nomeProprio ?nomeProprio .
       ?tipoO :tipoOficio ?tipoOficio .
       FILTER(?nomeProprio = "Martinus") .
}




SELECT ?hist ?doc
WHERE
{
   ?hist a :Historiador ; :controla ?doc .
}

SELECT ?nomeProprio ?tipoOficio
           WHERE { ?nomeProprio :tem_oficio ?tipoOficio }


SELECT ?nomeProprio ?tipoOficio
WHERE {
       ?nome :tem_oficio ?tipoO .
       ?nome :nomeProprio ?nomeProprio .
       ?tipoO :tipoOficio ?tipoOficio .
}

SELECT DISTINCT ?nomeProprio ?bem ?descricao
WHERE {
       ?nome :e_proprietario ?bem .
       ?nome :nomeProprio ?nomeProprio .
       ?bem :descricao ?descricao .
```

```
}

SELECT ?nomeClero ?nomePessoa
WHERE {
      ?nameC :tem_relacao ?nameP .
      ?nameC :nomeProprio ?nomeClero .
      ?nameP :nomeProprio ?nomePessoa .


}

SELECT ?testamento ?doacao ?bem ?desc
WHERE {
  {
    ?testamento :envolve_doacao ?doacao
  }

  {
   ?doacao :envolve_bem ?bem .
   ?bem :descricao ?desc
  }
}

SELECT DISTINCT ?Arquivo
WHERE {
       ?Documento :pertence_ao_arquivo ?Arquivo.
}

SELECT ?name ?testamento ?doacao ?bem ?desc
WHERE {
{
   ?clero :tem_testamento ?testamento.
   ?clero :nomeProprio ?name

}

 {
    ?testamento :envolve_doacao ?doacao
  }

  {
   ?doacao :envolve_bem ?bem .
   ?bem :descricao ?desc
  }
}


SELECT ?nomeProprio ?tipoOficio
WHERE {
      ?nome :tem_oficio ?tipoO .
```

```
      ?nome :nomeProprio ?nomeProprio .
      ?tipoO :tipoOficio ?tipoOficio .
      FILTER(?nomeProprio != "desconhecido")
}


SELECT ?nomeProprio ?funcao ?locate
WHERE {
      ?nome :tem_funcao ?func .
      ?nome :nomeProprio ?nomeProprio .
      ?func :funcaoEstatuto ?funcao .
      ?func :local ?locate .
      FILTER(?nomeProprio != "desconhecido" &&
          regex(?locate,"Porto","i"))

}


SELECT DISTINCT ?nomeproprioC ?tipoparentesco ?nomeproprioP
WHERE {
       {
       ?clero :tem_relacao_parente ?relacao.
       ?clero :nomeProprio ?nomeproprioC.
       ?relacao :tipoParentesco ?tipoparentesco.
       FILTER(?tipoparentesco = "Filho").

       }
       {
        ?relacao :e_pessoa_parente ?pessoa.
        ?pessoa :nomeProprio ?nomeproprioP.
        FILTER(?nomeproprioP != "desconhecido").
       }

}
```

Apêndice B

Código do Mapping

Neste apêndice apresenta-se o mapping criado para .....

```
[PrefixDeclaration]
:
    http://www.semanticweb.org/nelsonsa12/ontologies/2018/3/untitled-ontology-12#

[SourceDeclaration]
sourceUri  Clero
connectionUrl jdbc:mysql://localhost:3306/siepfinal
username root
password
driverClass com.mysql.jdbc.Driver

[MappingDeclaration] @collection [[
mappingId  Clero
target     :Clero#{CodClerigo} a :Clero ; :citado_em
    :Documento#{CotaDocumento} .
source     SELECT c.CodClerigo, d.CotaDocumento FROM clerocatedralicio
    AS c INNER JOIN clerodocumentos AS cd ON c.CodClerigo=cd.CodClerigo
    INNER JOIN documentos AS d ON d.CotaDocumento=cd.CotaDocumento

mappingId  Clero2
target     :Clero#{CodClerigo} a :Clero ; :e_proprietario :Bem#{CodBem} .
source     SELECT c.CodClerigo, b.CodBem FROM clerocatedralicio AS c
    INNER JOIN bens AS b ON c.CodClerigo=b.CodClerigo

mappingId  Clero3
target     :Clero#{CodClerigo} a :Clero ; :pertence_a
    :Ordem_Sacra#{CodOrdenacao} .
source     SELECT c.CodClerigo, os.CodOrdenacao FROM clerocatedralicio
    AS c INNER JOIN ordenssacras AS os ON c.CodClerigo=os.CodClerigo
```

```
mappingId  Clero4
target     :Clero#{CodClerigo} a :Clero ; :tem_formacao
    :Formacao#{CodFormacao} .
source     SELECT c.CodClerigo, f.CodFormacao FROM clerocatedralicio AS
    c INNER JOIN formacao AS f ON c.CodClerigo=f.CodClerigo

mappingId  Clero5
target     :Clero#{CodClerigo} a :Clero ; :tem_funcao
    :Funcao#{CodCleroFuncao} .
source     SELECT c.CodClerigo, cf.CodCleroFuncao FROM clerocatedralicio
    AS c INNER JOIN clerofuncoes AS cf ON c.CodClerigo=cf.CodClerigo

mappingId  Clero6
target     :Clero#{CodClerigo} a :Clero ; :tem_hieraquia
    :Hieraraquia#{CodCleroCat} .
source     SELECT c.CodClerigo, h.CodCleroCat FROM clerocatedralicio AS
    c INNER JOIN clerocathierarquica AS h ON c.CodClerigo=h.CodClerigo

mappingId  Clero7
target     :Clero#{CodClerigo} a :Clero ; :tem_oficio
    :Oficio#{CodOficio} .
source     SELECT c.CodClerigo, o.CodOficio FROM clerocatedralicio AS c
    INNER JOIN clerooficios AS o ON c.CodClerigo=o.CodClerigo

mappingId  Clero8
target     :Clero#{CodClerigo} a :Clero ; :tem_testamento
    :Testamento#{CodUltVontade} .
source     SELECT c.CodClerigo, t.CodUltVontade FROM clerocatedralicio
    AS c INNER JOIN ultimasvontades AS t ON c.CodClerigo=t.CodClerigo

mappingId  Clero9
target     :Clero#{CodClerigo} a :Clero ; :nomeProprio {NomeProprio} ;
    :observacoes {Observacoes} ; :patronimico {Patronimico} ;
    :prebendado {Prebendado} ; :terceiroElemento {TerceiroElemento} ;
    :tipoTerceiroElemento {TipoTerceiroElem} ; :localNascimento
    {LocalNasc} ; :localSepultura {LocalSepul} ; :dioceseNascimento
    {DioceseNasc} ; :dioceseSepultura {DioceseSepul} ; :data_de_obito
    {DataObito} .
source     SELECT CodClerigo, NomeProprio, Observacoes, Patronimico,
    Prebendado, TerceiroElemento, TipoTerceiroElem, LocalNasc,
    LocalSepul, DioceseNasc, DioceseSepul, DataObito FROM
    clerocatedralicio

mappingId  Documento
target     :Documento#{CotaDocumento} a :Documento ;
    :pertence_ao_arquivo :Arquivo#{Arquivo} ; :e_controlado
    :Historiador#{historiador} .
source     SELECT CotaDocumento, historiador, Arquivo FROM documentos

mappingId  Documento2
```

```
target     :Documento#{CotaDocumento} a :Documento ; :dataTopica
    {DataTopica} ; :eclesioasticosRef {EclesiasticosRef} ; :entEmiss
    {EntEmiss} ; :fundoTitulo {FundoTitulo} ; :maco {Maco} ;
    :numClerigos {NumClerigos} ; :numFolio {NumFolio} ; :observacoes
    {Observacoes} ; :publicado {Publicado} ; :sumario {Sumario} ;
    :sinaisVal {SinaisVal} ; :localRedigido {Localred} ;
    :data_de_criacao {DataCron} .
source     SELECT CotaDocumento, DataTopica, EclesiasticosRef, EntEmiss,
    FundoTitulo, Maco, NumClerigos, NumFolio, Observacoes, Publicado,
    Sumario, SinaisVal, LocalRed, DataCron FROM documentos

mappingId  Bem
target     :Bem#{CodBem} a :Bem ; :data_de_aquisacao {DataAquisicao} ;
    :descricao {Descricao} ; :observacoes {Observacoes} ; :rendaPreco
    {RendaPreco} ; :tipoBem {TipoBem} ; :tipoTransacao {TipoTransaccao}
    ; :local {Local} .
source     SELECT CodBem, DataAquisicao, Descricao, Observacoes,
    RendaPreco, TipoBem, TipoTransaccao, Local FROM bens

mappingId  Colacao
target     :Colacao#{CodColacao} a :Colacao ; :catEclesiastica
    {CatEclesiastica} ; :igreja {Igreja} ; :diocese {Diocese} .
source     SELECT CodColacao, CatEclesiastica, Igreja, Diocese FROM
    colacoes

mappingId  Fundacao
target     :Fundacao#{CodFund} a :Fundacao ; :tipoFundacao
    {TipoFundacao} ; :observacoes {Observacoes} ; :local {Local} .
source     SELECT CodFund, TipoFundacao, Observacoes, Local FROM
    ultimasvontadesfundacoes

mappingId  Oficio
target     :Oficio#{CodOficio} a :Oficio ; :tipoOficio {TipoOficio} .
source     SELECT CodOficio, TipoOficio FROM clerooficios

mappingId  Funcao
target     :Funcao#{CodCleroFuncao} a :Funcao ; :funcaoEstatuto
    {FuncaoEstatuto} ; :local {Local} .
source     SELECT CodCleroFuncao, FuncaoEstatuto, Local FROM clerofuncoes

mappingId  Formacao
target     :Formacao#{CodFormacao} a :Formacao ; :grau {Grau} ; :curso
    {Curso} ; :universidade {Universidade} .
source     SELECT CodFormacao, Grau, Curso, Universidade FROM formacao

mappingId  Ordem_Sacra
target     :Ordem_Sacra#{CodOrdenacao} a :Ordem_Sacra ; :ordemSacra
    {OrdemSacra} ; :diocese {Diocese} ; :igreja {Igreja} .
source     SELECT CodOrdenacao, OrdemSacra, Diocese, Igreja FROM
    ordenssacras
```

```
mappingId  Pessoa
target     :Pessoa#{CodIndRelacionado} a :Pessoa ; :e_dependente
    :Clero#{CodClerigo} .
source     SELECT p.CodIndRelacionado, c.CodClerigo FROM
    clerocatedralicio AS c INNER JOIN clerodependentes AS cp ON
    c.CodClerigo = cp.CodClerigo INNER JOIN individuosrelacionados AS p
    ON cp.CodIndRelacionado = p.CodIndRelacionado

mappingId  Pessoa2
target     :Pessoa#{CodIndRelacionado} a :Pessoa ; :tem_oficio
    :Oficio#{CodOficio} .
source     SELECT p.CodIndRelacionado, o.CodOficio FROM
    individuosrelacionados AS p INNER JOIN clerooficios as o ON
    o.CodIndRelacionado = p.CodIndRelacionado

mappingId  Pessoa3
target     :Pessoa#{CodIndRelacionado} a :Pessoa ; :nomeProprio
    {NomeProprio} ; :observacoes {Observacoes} ; :patronimico
    {Patronimico} ; :terceiroElemento {TerceiroElemento} ;
    :tipoTerceiroElemento {TipoTerceiroElemento} ; :estatuto {Estatuto}
    ; :localFuncao {LocalFuncao} .
source     SELECT NomeProprio, Observacoes, Patronimico,
    TerceiroElemento, TipoTerceiroElemento, Estatuto, LocalFuncao FROM
    individuosrelacionados

mappingId  Historiador
target     :Historiador#{historiador} a :Historiador ; :controla
    :Documento#{CotaDocumento} .
source     SELECT historiador, CotaDocumento FROM documentos

mappingId  Historiador2
target     :Historiador#{historiador} a :Historiador ; :estuda_clero
    :Clero#{CodClerigo} .
source     SELECT historiador, CodClerigo FROM clerocatedralicio

mappingId  Testamento
target     :Testamento#{CodUltVontade} a :Testamento ; :envolve_fundacao
    :Fundacao#{CodFund} .
source     SELECT t.CodUltVontade, f.CodFund FROM ultimasvontades AS t
    INNER JOIN ultimasvontadesfundacoes AS f ON t.CodUltVontade =
    f.CodUltVont

mappingId  Testamento2
target     :Testamento#{CodUltVontade} a :Testamento ; :envolve_doacao
    :Doacao#{CodDoacao} .
source     SELECT t.CodUltVontade, d.CodDoacao FROM ultimasvontades AS t
    INNER JOIN ultimasvontadesbens AS d ON t.CodUltVontade =
    d.CodUltVontade
```

```
mappingId  Testamento3
target     :Testamento#{CodUltVontade} a :Testamento ; :tem_testemunha
    :Pessoa#{CodIndRelacionado} .
source     SELECT t.CodUltVontade, p.CodIndRelacionado FROM
    ultimasvontades AS t INNER JOIN individuosrelacionados AS p ON
    t.CodIndRelTest = p.CodIndRelacionado

mappingId  Testamento4
target     :Testamento#{CodUltVontade} a :Testamento ; :tem_herdeiro
    :Pessoa#{CodIndRelacionado} .
source     SELECT t.CodUltVontade, p.CodIndRelacionado FROM
    ultimasvontades AS t INNER JOIN individuosrelacionados AS p ON
    t.CodIndRelRed = p.CodIndRelacionado

mappingId  Testamento5
target     :Testamento#{CodUltVontade} a :Testamento ; :data_de_feitura
    {DataFeitura} ; :situacaoDitou {SituacaoDitou} ; :local {Local} .
source     SELECT DataFeitura, SituacaoDitou, Local FROM ultimasvontades

mappingId  Doacao
target     :Doacao#{CodDoacao} a :Doacao ; :envolve_bem :Bem#{CodBem} .
source     SELECT d.CodDoacao, b.CodBem FROM ultimasvontadesbens AS d
    INNER JOIN bens AS b ON d.BensDoados = b.CodBem

mappingId  Doacao2
target     :Doacao#{CodDoacao} a :Doacao ; :nomeInstituicaoBenef
    {NomeInstitBenef} ; :nomeDonatario {NomeDonatario} .
source     SELECT NomeInstitBenef, NomeDonatario FROM ultimasvontadesbens

mappingId  Hierarquia
target     :Hierarquia#{CodCleroCat} a :Hierarquia ; :catEclesiastica
    {CatEclesiastica} ; :data_inicio {DataInicio} ; :data_fim {DataFim}
    ; :igreja {Igreja} ; :diocese {Diocese} ; :causaSaida {CausaSaida} .
source     SELECT Cateclesiastica, DataInicio, DataFim, Igreja, Diocese,
    Causasaida FROM clerocathierarquica

mappingId  Documento3
target     :Documento#{CotaDocumento} a :Documento ; :tem_citacao
    :Clero#{CodClerigo} .
source     SELECT d.CotaDocumento, c.CodClerigo FROM clerocatedralicio
    AS c INNER JOIN clerodocumentos AS cd ON c.CodClerigo=cd.CodClerigo
    INNER JOIN documentos AS d ON d.CotaDocumento=cd.CotaDocumento

mappingId  Clero10
target     :Clero#{CodClerigo} a :Clero ; :tem_dependente
    :Pessoa#{CodIndRelacionado} .
source     SELECT c.CodClerigo, p.CodIndRelacionado FROM
    clerocatedralicio AS c INNER JOIN clerodependentes AS cp ON
    c.CodClerigo = cp.CodClerigo INNER JOIN individuosrelacionados AS p
    ON cp.CodIndRelacionado = p.CodIndRelacionado
```

```
mappingId  Clero11
target     :Clero#{CodClerigo} a :Clero ; :tem_relacao
    :Pessoa#{CodIndRelacionado} .
source     SELECT c.CodClerigo, p.CodIndRelacionado FROM
    clerocatedralicio AS c INNER JOIN cleroparentes AS cp ON
    c.CodClerigo = cp.CodClerigo INNER JOIN individuosrelacionados AS p
    ON cp.CodIndRelacionado = p.CodIndRelacionado

mappingId  Pessoa4
target     :Pessoa#{CodIndRelacionado} a :Pessoa ; :tem_relacao
    :Clero#{CodClerigo} .
source     SELECT p.CodIndRelacionado, c.CodClerigo FROM
    clerocatedralicio AS c INNER JOIN cleroparentes AS cp ON
    c.CodClerigo = cp.CodClerigo INNER JOIN individuosrelacionados AS p
    ON cp.CodIndRelacionado = p.CodIndRelacionado

mappingId  Clero12
target     :Clero#{CodClerigo} a :Clero ; :tem_clero_dependencia
    :Clero_Dependencia#{CodCleroDependente} .
source     SELECT c.CodClerigo, d.CodCleroDependente FROM
    clerocatedralicio AS c INNER JOIN clerodependentes AS d ON
    c.CodClerigo=d.CodClerigo

mappingId  Clero14
target     :Clero#{CodClerigo} a :Clero ; :tem_posicao_documento
    :Posicao_Clero_Doc#{Codclerodocs} .
source     SELECT c.CodClerigo, d.Codclerodocs FROM clerocatedralicio AS
    c INNER JOIN clerodocumentos AS d ON c.CodClerigo=d.CodClerigo

mappingId  Clero_Depend
target     :Clero_Dependencia#{CodCleroDependente} a :Clero_Dependencia
    ; :e_pessoa_dependente :Pessoa#{CodIndRelacionado} .
source     SELECT c.CodCleroDependente, p.CodIndRelacionado FROM
    clerodependentes AS c INNER JOIN individuosrelacionados AS p ON
    c.CodIndRelacionado=p.CodIndRelacionado

mappingId  Clero_Depend2
target     :Clero_Dependencia#{CodCleroDependente} a :Clero_Dependencia
    ; :tipoDependencia {TipoDependencia} .
source     SELECT TipoDependencia FROM clerodependentes

mappingId  Relacao_Parente
target     :Relacao_Parente#{CodParente} a :Clero_Dependencia ;
    :e_pessoa_parente :Pessoa#{CodIndRelacionado} .
source     SELECT c.CodParente, p.CodIndRelacionado FROM cleroparentes
    AS c INNER JOIN individuosrelacionados AS p ON
    c.CodIndRelacionado=p.CodIndRelacionado

mappingId  Relacao_Parente2
```

```
target     :Relacao_Parente#{CodParente} a :Relacao_Parente ;
    :tipoParentesco {TipoParentesco} .
source     SELECT TipoParentesco FROM cleroparentes

mappingId  Posicao_Clero
target     :Posicao_Clero_Doc#{Codclerodocs} a :Posicao_Clero_Doc ;
    :refere_a_documento :Documento#{CotaDocumento} .
source     SELECT c.Codclerodocs, d.CotaDocumento FROM clerodocumentos
    AS c INNER JOIN documentos AS d ON c.CotaDocumento=d.CotaDocumento

mappingId  Posicao_Clero2
target     :Posicao_Clero_Doc#{Codclerodocs} a :Posicao_Clero_Doc ;
    :posicaoCleroDoc {PosicaoCleroDoc} .
source     SELECT PosicaoCleroDoc FROM clerodocumentos

mappingId  Clero13
target     :Clero#{CodClerigo} a :Clero ; :tem_relacao_parente
    :Relacao_Parente#{CodParente} .
source     SELECT c.CodClerigo, cp.CodParente FROM clerocatedralicio AS
    c INNER JOIN cleroparentes AS cp ON c.CodClerigo=cp.CodClerigo
]]
```