



Universidade do Minho

Mestrado Integrado em Engenharia Informática

Processamento e Representação de Conhecimento

Contratos de concurso público portugueses

Alexandre Dias (A78425)

14 de Junho de 2019

Conteúdo

1	Introdução	1
2	Dataset	1
2.1	Ontologia Criada	2
3	API da aplicação	5
4	Interface da aplicação	5
5	Conclusão	6

Lista de Figuras

1 Introdução

Este projeto foi realizado no âmbito de Processamento e Representação de Conhecimento, com o intuito de tirar partido do grande volume de dados disponível publicamente.

Neste irá ser feita a análise da estrutura e processamento de dados para permitir o seu armazenamento numa base de dados, bem como permitir consultas e formas intuitivas de apresentação dos mesmos. Para tal irá ser utilizada a ferramenta GraphDB para a base de dados e o NodeJS para permitir a criação de uma aplicação que consiga realizar interrogações à base de dados e expor a informação por ela devolvida.

2 Dataset

Para a realização desta aplicação foi utilizado um dataset contendo contratos de concurso público, este foi retirado do sítio governamental, *www.base.gov.pt*, sendo que este foi exportado inicialmente como um ficheiro em formato CSV.

Este foi inicialmente convertido para formato JSON com recurso a um conversor online sendo que este inicialmente ficou com a seguinte estrutura:

```
{
  "Objeto do Contrato": "12/100/2019 - Aquisio de Material p/Preparao
    Citosttticos",
  "Tipo de Procedimento": "Concurso pblico",
  "Tipo(s) de Contrato": "Aquisio de bens mveis",
  "CPVs": "33140000-3, Material mdico de consumo",
  "Entidade(s) Adjudicante(s)": "Centro Hospitalar Universitrio do
    Algarve, E.P.E. (510745997)",
  "Entidade(s) Adjudicatria(s)": "B.BRAUN MEDICAL LDA (501506543),
    Fresenius Kabi Pharma Portugal, Lda. (504293753), VYGON (PORTUGAL
    ), LDA (503232785), PMH - PRODUTOS MDICOS-HOSPITALARES, S.A.
    (502376899), BARNA IMPORT MEDICA PORTUGAL, UNIPessoal LDA
    (513468994), CLINIFAR SA (503281115), Medline International
    Portugal Unipessoal, Lda (510533515), CODAN 11- VENDA E
    DISTRIBUIO DE INSTRUMENTPOS MDICOS E ACESSRIOS S.A. (504923471)",
  "Preo Contratual": "273.698,43 ",
  "Data de Publicao ": "24-04-2019",
  "Data de Celebrao do Contrato": "23-04-2019",
  "Prazo de Execuo ": "618 dias",
  "Local de Execuo ": "Portugal, Faro<BR/>Portugal",
  "Fundamentao ": "Artigo 20. , n. 1, alnea a) do Cdigo dos
    Contratos Pblicos",
  "Data de Fecho do Contrato": "",
  "Preo Total Efetivo": "",
  "Causas das Alteraes ao Prazo": "",
  "Causas das Alteraes ao Preo ": "",
  "Estado": "",
  "N. registo do Acordo Quadro": "No aplicvel.",
```

```

    "Descrição do Acordo Quadro": "Não aplicável.",
    "Procedimento Centralizado": false
  }

```

Esta estrutura foi então utilizada para criar uma ontologia, e posteriormente foi realizado o processamento dos dados para converter o formato JSON, para um formato *turtle*, que permitiu a importação da ontologia para o *GraphDB*, a base de dados utilizada para a aplicação.

2.1 Ontologia Criada

Dada a estrutura do dataset, foi criada uma ontologia com as seguintes classes:

- Entidade
 - Esta classe irá representar as entidades dos contratos.
- Contrato
 - Esta classe irá ser representada para representar os contratos, possuindo as seguintes subclasses:
 - AquisicaoBensMoveis
 - Esta classe irá ser utilizada para representar os contratos com o tipo aquisição de bens móveis.
 - AquisicaoServicos
 - Esta classe irá ser utilizada para representar os contratos com o tipo aquisição de serviços.
 - ConcessaoObrasPublicas
 - Esta classe irá ser utilizada para representar os contratos com o tipo concessão de obras públicas.
 - ConcessaoServicosPublicos
 - Esta classe irá ser utilizada para representar os contratos com o tipo concessão de serviços públicos.
 - EmpreitadasObrasPublicas
 - Esta classe irá ser utilizada para representar os contratos com o tipo empreitadas de obras públicas.
 - LocacaoBensMoveis
 - Esta classe irá ser utilizada para representar os contratos com o tipo locação de bens móveis.
 - Sociedade
 - Esta classe irá ser utilizada para representar os contratos com o tipo Sociedade.
 - Outros
 - Esta classe irá ser utilizada para representar os contratos com outros tipos.

Além destas classes, foram ainda geradas para a ontologia algumas propriedades de objetos, bem como algumas propriedades de dados:

- eAdjudicanteDe
Propriedade entre objetos, possuindo o domínio de entidade e contra-domínio de contratos. Será utilizada para caracterizar a relação "*é entidade adjudicante de*". É a propriedade inversa de entidadeAdjudicante.
- eAdjudicatariaDe
Propriedade entre objetos, possuindo o domínio de entidade e contra-domínio de contratos. Será utilizada para caracterizar a relação "*é entidade adjudicatária de*". É a propriedade inversa de entidadeAdjudicataria.
- entidadeAdjudicante
Propriedade entre objetos, possuindo o domínio de contrato e contra-domínio de entidade. Será utilizada para caracterizar a relação "*tem a seguinte entidade adjudicante*".
- entidadeAdjudicataria
Propriedade entre objetos, possuindo o domínio de contrato e contra-domínio de entidade. Será utilizada para caracterizar a relação "*tem a seguinte entidade adjudicatária*".
- causasAlteracoesPrazo
Propriedade de dados, utilizada para indicar as causas de alteração do prazo de um contrato.
- causasAlteracoesPreco
Propriedade de dados, utilizada para indicar as causas de alteração do preço de um contrato.
- cpv
Propriedade de dados, utilizada para indicar o CPV de um contrato.
- dataCelebracaoContrato
Propriedade de dados, utilizada para indicar a data de celebração de um contrato.
- dataFechoContrato
Propriedade de dados, utilizada para indicar a data de fecho de um contrato.
- dataPublicacao
Propriedade de dados, utilizada para indicar a data de publicação de um contrato.

- `descricaoAcordoQuadro`
Propriedade de dados, utilizada para indicar a descrição de acordo de quadro de um contrato.
- `estado`
Propriedade de dados, utilizada para indicar o estado de um contrato.
- `fundamentacao`
Propriedade de dados, utilizada para indicar a fundamentação de um contrato.
- `localExecucao`
Propriedade de dados, utilizada para indicar o local de execução de um contrato.
- `numeroRegistoAcordoQuadro`
Propriedade de dados, utilizada para indicar o número de registo do acordo quadro de um contrato.
- `objetoContrato`
Propriedade de dados, utilizada para indicar o objeto do contrato.
- `prazoExecucao`
Propriedade de dados, utilizada para indicar o prazo de execução de um contrato.
- `precoContratual`
Propriedade de dados, utilizada para indicar o preço contratual de um contrato.
- `procedimentoCentralizado`
Propriedade de dados, utilizada para indicar se um contrato será um procedimento centralizado ou não, esta possui apenas dois valores possíveis: `true` e `false`. Tendo um contradomínio do tipo booleano.
- `causasAlteracoesPreco`
Propriedade de dados, utilizada para indicar as causas de alteração do preço de um contrato.
- `outrosDesc`
Propriedade de dados poderá ocorrer apenas em contratos que possuam outros tipos enquanto um dos tipos do contrato, fornecendo uma breve descrição do motivo alternativo.

Para facilitar e melhorar o processamento dos dados, inicialmente os tipos de contrato foram convertidos para uma lista com recurso a um programa de `awk` e um *script* de `python` para realizar uma formatação inicial do ficheiro. Tal foi possível uma vez que contratos com múltiplos tipos possuíam os diferentes tipos divididos através da marca `
`.

Posteriormente foi utilizado um programa realizado com recurso a `NodeJS` que iria converter o ficheiro `JSON` para triplos de `OWL` que seguissem a especificação da ontologia anteriormente desenvolvida.

Posteriormente alguns erros foram manualmente corrigidos, tendo sido apagados alguns caracteres desformatados ou corrompidos, bem como apagadas algumas aspas pontuais que ocorriam nas descrições do objeto de contrato, uma vez que, embora estas não afetassem diretamente o processamento do ficheiro `JSON`, após uma conversão inicial para um ficheiro *turtle*, estas tinham sido originadoras de erros detetados pelo *Raptor*.

Ainda assim, esta conversão para *turtle* mostrou-se não completamente correta, devido a uma falta de estrutura e de um separador claro na apresentação das entidades que dificultou o seu processamento, o que deu origem a alguns erros na geração das entidades na ontologia.

3 API da aplicação

Para realizar interrogações à base de dados, foi criado um servidor com recurso ao `express`. Este possuirá um conjunto de rotas predefinidas que realizaram interrogações à base de dados e retornarão a resposta. Este servidor foi criado como forma de facilitar a utilização da base de dados, oferecendo um conjunto de interrogações possíveis à mesma nas suas rotas.

Este servidor irá responder apenas com dados no formato `JSON`, pelo que esta não possui nenhuma interface gráfica para a sua utilização.

4 Interface da aplicação

Com o recurso ao `Vue`, foi criado um servidor que seria responsável por permitir a navegação entre páginas e fornecer um meio de expor os dados que seriam acessíveis no servidor da API da aplicação. Neste foi também utilizado o módulo `Vuetify` do `Vue`.

Para esta aplicação foram criadas apenas 4 páginas, sendo que estas permitiriam:

- Consultar todos os contratos
- Consultar todas as entidades
- Consultar um contrato
- Consultar uma entidade

Sendo que em todas elas foram utilizadas foram utilizadas primitivas do módulo Vuetify para permitir a geração de botões para a navegação entre páginas e de tabelas para a apresentação dos dados.

5 Conclusão

Através desta aplicação foi possível perceber melhor as vantagens e dificuldades de utilização e modelação de dados disponíveis publicamente.

Futuramente poderiam ser implementadas mais rotas, uma vez que nem todas as rotas de interrogações de dados existentes na API serão utilizadas pelo servidor responsável pela exposição de dados, bem como ser melhorada a interface da aplicação.