

MIECT

Engenharia de Dados e Conhecimento

Ano Letivo 2019/2020

Realizado por:

Gabriel Silva nºMec: 85129

Gonçalo Vitor nºMec: 85119

João Monteiro nºMec: 80272

Lucas Seabra nºMec: 80323



universidade
de aveiro

Introdução

Aplicando os conhecimentos e as tecnologias que nos foram introduzidas no âmbito da cadeira de Engenharia de Dados e Conhecimento, e no contexto do segundo projeto da referente cadeira, decidimos desenvolver uma aplicação amigável ao utilizador que permitisse visualizar dados atualizados referentes a países, possibilitando ao utilizador pesquisar diferentes países.

Para tal possuímos uma base de dados com países e capitais. Através desta o utilizador pode visualizar em diferentes países: área; população; bandeira e até mesmo comparar crescimento populacional e ou económico.





















É de notar ainda que a aplicação possui autenticação de utilizadores permitindo assim acesso exclusivo e também a possibilidade de adicionar notas.

HOME

NOTAS

Search for country name or plate

LOGOUT

	PLATE	NAME	LOCALTIME	CURRENCY	POPULATION	AREA (KM2)	INFLATION (%)	PIB (%)
	DZ	Algeria	UTC+01:00	DZD	35591377	2381741	5.9	4.2
		Angola	UTC+01:00	AOA	29784193	1246700	41.9	0
		Antigua Barbuda	UTC-04:00	XCD	100963	440.29	-0.6	3.7
	RA	Argentina	UTC-03:00	ARS	44938712	2780400		-2.3
		Artsakh	UTC+04:00	AMD	146600	11458.1074		
	AUS	Australia	UTC+10:00	AUD	23401892	7692024	2.1	2.5
	A	Austria	UTC+01:00	EUR	8809212	83878.99	1.5	1.5
	BS	Bahamas	UTC-05:00	BSD	377374	13878	0.4	0
	BD	Bangladesh		BDT	156594962	147570	6.2	6.9
	BDS	Barbados	UTC-04:00	BBD	284644	439	0.8	1.6
	DY	Benin	UTC+01:00	XOF	10008749	114763	-2.7	4
	BOL	Bolivia	UTC-04:00	BOB	11051600	1098581	4	4.1
	BW	Botswana	UTC+02:00	BWP	2021144	581737	3	2.9
	BR	Brazil	UTC-02:00	BRL	208494900	8515767	6.3	-3.6
	BRU	Brunei	UTC+08:00	BND	417784	5765.313533	-1.6	-3.2
	CI	Bte Ivoire	UTC±00:00	XOF	20316086	322463	1.2	7.5
	BG	Bulgaria	UTC+02:00	BGN	7265115	110993.6	-0.5	3.4
	RU	Burundi	UTC+02:00	BIF	8053574	27834	9.5	-1
	K	Cambodia	UTC+07:00	KHR	15135169	181035	3.5	7
	CAM	Cameroon	UTC+01:00	XAF	17463836	475442	0.3	4.4

123...8

At page 1 showing 20 of 142 countries.

A nossa página inicial.


HOME

NOTAS

Search for country name or plate

LOGOUT

Portugal




Inflation: 0.9%

Pib: 1.4%


Area: 92212Km2

Population: 10600000

Life: 81.12683 Years




Capital - Lisboa



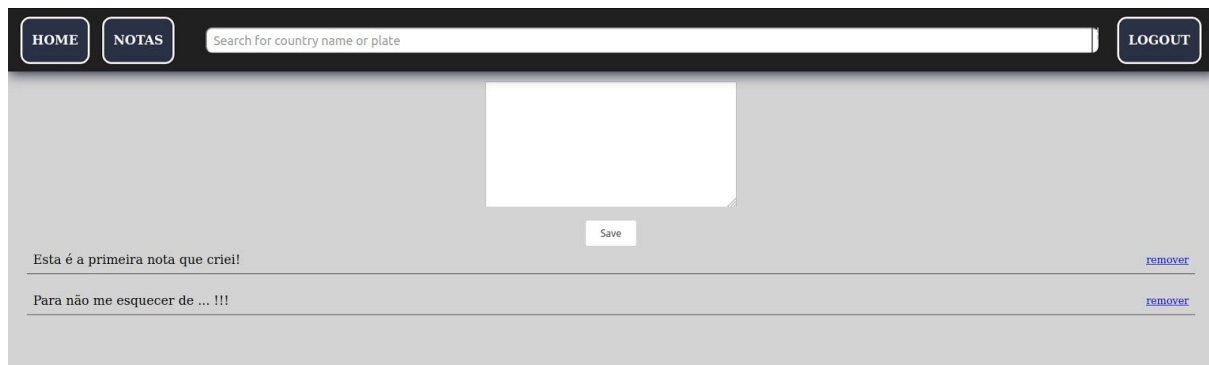
Area: 100.05 Km2

Population: 506654

President - Marcelo Rebelo de Sousa



A página relativa um país específico.



HOME NOTAS Search for country name or plate LOGOUT

Save

Esta é a primeira nota que criei! [remover](#)

Para não me esquecer de ... !!! [remover](#)

A páginas das notas.

Dados, suas fontes e sua transformação

De modo a satisfazer a necessidade de dados no formato RDF, referente a países e devido ao facto de não termos encontrado dados no formato RFD desejável, neste caso em N3, criamos um script que faz pedidos à API da Wikidata por forma a guardarmos os dados fornecidos pela mesma na nossa base dados, permitindo assim atualização de dados na nossa base dados. Optámos por criar a nossa base de dados assim, porque depois de termos arranjado um ficheiro em N3 capaz de suportar o nosso projeto com um bom volume dados, deparámo-nos com várias dificuldades principalmente relacionadas com a discrepância do formato dos dados do nosso ficheiro e da Wikidata. Isto tornava a obtenção de dados em *real time* da Wikidata muito trabalhoso e custoso, dada a dificuldade de associar um país na nossa base de dados com o código respectivo na Wikidata.

A obtenção destes dados para a nossa base de dados é feita pelo ficheiro 'update_bd.py'. No ficheiro podemos ver múltiplas *queries* à Wikidata seguidas de *updates* na nossa base de dados.

Operações sobre os dados (SPARQL)

Todas as operações de pesquisa ou inserção de dados são efetuadas através de SPARQL.

As primeiras queries que vamos descrever são as presentes no ficheiro *update_bd.py*. Como referido acima, o propósito deste ficheiro é fazer pesquisas na Wikidata e de seguida atualizar a nossa base de dados.

A primeira pesquisa feita é relativa a países. Nesta pesquisa, para além de procurarmos por todos os países existentes na Wikidata, extraímos também, se existente, informação sobre a abreviação do nome do país (e.g. United States → USA), a hora do país (UTC +/- *valor*), a sua população, esperança média de vida, localização geográfica, inflação, área em km², moeda usada, o PIB, a sua capital e o presidente. Segue-se a esta query um update que insere todos os países na nossa base de dados.

A seguir fazemos algo semelhante, mas para as capitais. Primeiro fazemos uma pesquisa onde procuramos, para a capital de cada país, uma imagem sua, o seu nome, a

população e área em km². Depois, tal como antes, adicionamos todas as capitais à nossa base de dados.

Depois, fazemos o mesmo para os presidentes. Procuramos na Wikidata o seu nome e uma foto e inserimos na nossa base de dados.

Seguem-se agora as queries feitas pela aplicação. Começaremos pelas da página inicial onde se faz o login/registo.

O login faz uma query muito simples, que simplesmente procura na base de dados se existe algum user com o mesmo nome e a mesma password. Se sim, o login é feito com sucesso, caso contrário há uma mensagem de erro. O registo envolve duas queries: a primeira verifica se o nome do user já existe na base de dados, pois não pode haver usernames replicados; a segunda faz uma inserção do nome e da password. Os utilizadores são inseridos com dois tipos, user e Person (este último da ontologia conhecida FOAF).

O grupo decidiu adicionar também uma funcionalidade de notas no projeto. No cimo da página das notas há um janela onde podemos escrever e um botão para adicionar a nota à base de dados, o que é feito por uma query. Nessa mesma página é feita, inicialmente, uma query que procura todas as notas do utilizador que está logged in na aplicação e as mostra por baixo da janela. As notas mostradas têm um link para remover a própria nota. Essa remoção é feita pela nossa última query relativa às notas.

Gostaríamos de referir que as funcionalidades descritas acima e mencionadas no fim da introdução, nomeadamente o login/registo e a criação de notas, apesar de não se inserirem no tema/serem desnecessárias, foram criadas para que fizéssemos um uso mais alargado desta tecnologia de manipulação de dados. Não havendo estas funcionalidades, não haveria *updates* dos dados da nossa aplicação devido à interação do utilizador com a aplicação.

Faltam apenas as queries feitas na página principal e na página de um país específico.

Na página principal apresentamos uma tabela paginada, isto é, que mostra apenas vinte resultados de cada vez, mas que tem várias páginas, todas elas mostrando vinte resultados, até à última, que pode mostrar vinte, no caso do número de países ser múltiplo de vinte, ou menos. Também para fazer isto foram precisas queries. Primeiro, fazemos uma query que procura todos os países. O interesse desta query é obter a quantidade de países para sabermos quantos países há na página final, visto que mostramos essa informação ao utilizador em todas as páginas. E depois temos a query que vai buscar todos os dados dos países. Esta pesquisa é muito semelhante à que é feita no ficheiro *'update_bd.py'*, no entanto tem alguns detalhes a mais para possibilitar a paginação, nomeadamente um limite de resultados devolvidos e um offset para que as páginas tenham todas os dados que lhe correspondem e ainda a ordenação dos dados por qualquer tipo de dados mostrados (as diferentes colunas da tabela) e em qualquer direção (ascendente e descendente). Esta última parte, referente à ordenação, é controlada pelo utilizador.

Finalmente, temos a query da página de um país. Esta query faz uma pesquisa de todos os dados existentes acerca de um país em específico, da sua capital e do seu presidente.

Para além destas queries temos ainda as que usamos para inserir dados da Wikidata em *real time* na aplicação, mas falaremos sobre elas um pouco mais à frente.

A publicação de dados semânticos através de RDFa

Quanto à web semântica, utilizámos o conjunto de extensões e atributos RDFa no html, como se pode comprovar nas seguintes imagens:

The screenshot shows a web application with a table of countries. The table has columns: PLATE, NAME, LOCALITM, CURRENC, POPULATION, AREA (KM2), and INFLATI. The first row is for Algeria (DZ). The table is paginated, showing 1 to 8 of 142 countries. The browser's developer tools are open, showing the HTML source code with RDFa annotations. The annotations include `<rdf:type>`, `<rdf:property>`, `<rdf:resource>`, and `<rdf:literal>` tags, which are used to link the data to external RDF vocabularies like `http://www.wikidata.org`.

The screenshot shows a web application displaying detailed information for Angola. The page includes the national flag, a map of Angola, and key statistics: Inflation: 41.9%, Pib: 0%, Area: 1246700Km2, Population: 29784193, Life: 61.547 Years. It also lists the Capital - Luanda, the President - João Manuel Gonçalves Lourenço, and the Area: 113000000 Km2, Population: 2487444. The browser's developer tools are open, showing the HTML source code with RDFa annotations. The annotations include `<rdf:type>`, `<rdf:property>`, `<rdf:resource>`, and `<rdf:literal>` tags, which are used to link the data to external RDF vocabularies like `http://www.wikidata.org`.

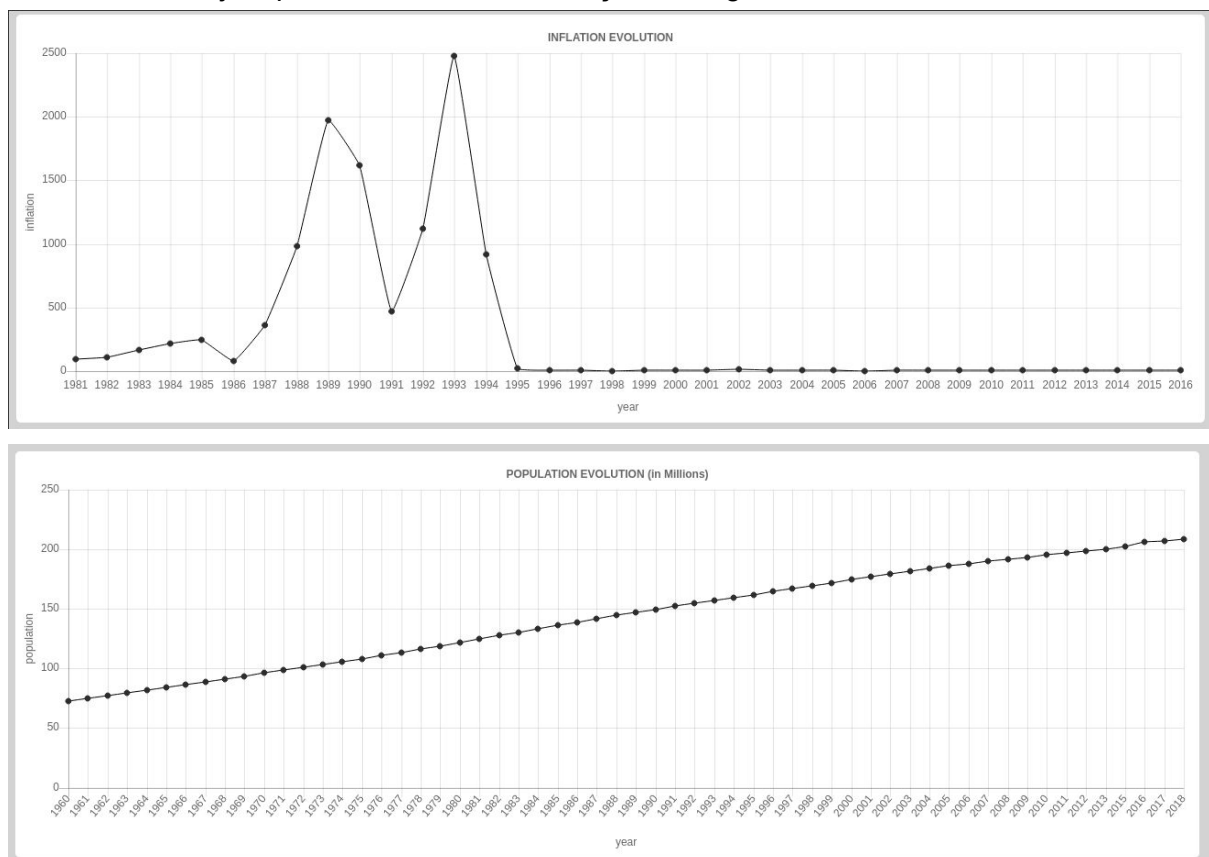
Note-se as múltiplas tags usadas nos vários elementos HTML. (no caso de não se verem bem as imagens, fazer zoom à página deve ajudar)

A integração de dados da Wikidata

Como foi referido anteriormente toda a nossa base de dados possui dados da Wikidata permitindo-nos ter acesso a dados atualizados. Podemos assim, afirmar que a integração dos dados da Wikidata foi fulcral para nossa aplicação.

A interação com a Wikidata para obtenção de dados em *real time* é feita na página de um país específico. A aplicação faz queries para obter dados relativos à inflação e à população de cada ano, desde 1980. Podemos assim ver uma evolução destes aspectos ao longo do tempo.

Os gráficos em si, foram construídos com recurso a uma biblioteca externa, chamada Chart.js, que facilita muito a criação dos gráficos, usando os nossos dados.



Evolução da inflação(em cima) e população(em baixo) do Brasil desde 1980.

Funcionalidades da Aplicação (UI)

A UI está desenvolvida de forma a respeitar os standards de uma aplicação *User Friendly*, o que a torna minimalista e de fácil aprendizagem. Para iniciar a aplicação é necessário um registo/login.

Funcionalidades:

- **Visualização de dados:**
 - Nomes de países
 - Bandeiras
 - Inflação
 - Área
 - População
 - PIB
 - Esperança média de vida
 - Localização no globo
 - Capital respetiva, sua área e população
 - Presidente
- **Edição de dados:**
 - Adicionar:
 - Users
 - Notas
 - Remover:
 - Notas
- Login/Registo
- Manipulação da tabela na página principal
 - Ordenação por qualquer tipo de dados, tanto ascendente como descendente
 - Pesquisa de países pelo nome ou plate (diminutivo do nome)

Conclusões

À semelhança do projeto anterior, encontrar um *dataset* que se aplicasse às nossas necessidades revelou-se um problema. E mesmo quando arranjámos um, acabámos por nos deparar com mais problemas e dificuldades e acabámos por recorrer à Wikidata para criar a nossa base de dados. Para além disso, o tema em si que escolhemos não dava muito espaço para integração de funcionalidades que permitissem edição e adição de dados que fizessem sentido. No entanto, dado que estas operações eram requisitos do projeto, o grupo adicionou funcionalidades que fizessem essas mesmas operações, ainda que essas funcionalidades não se enquadrassem muito bem no tema.

Em suma, o grupo acha que construiu uma aplicação funcional, e que, apesar das adversidades, acima de tudo, cumpre com os requisitos exigidos na descrição do projeto.

Configuração para executar a aplicação

Preparação para correr a aplicação:

Ligar a aplicação do graphDB;

Criar um repositório com o ID: edc_2019 e Base URL: http://edc_2019.org/ ;

De seguida correr o ficheiro `update_bd.py` (python3.7 `update_bd.py`), que está em *diretório_base_do_projeto/appl/* .

Executar a aplicação:

-> Certificar que a aplicação GraphDB está a correr e o repositório certo está a ser usado.

-> `python3.7 manage.py runserver` (num terminal)

ou executar no PyCharm

Por razões que o grupo foi incapaz de apurar, o login e registo só funcionam quando se usa o URL `http://localhost:8000/` , pelo que não se deve usar a alternativa `http://127.0.0.1:8000/`.

A aplicação foi construída em linux. O projeto tem um ficheiro `requirements.txt` que deve facilitar a instalação de todas as dependências necessárias para correr a aplicação.

Nota

Todas as imagens deste relatório encontram-se no .zip no qual este mesmo foi enviado, caso a visualização das mesmas esteja dificultada aqui.