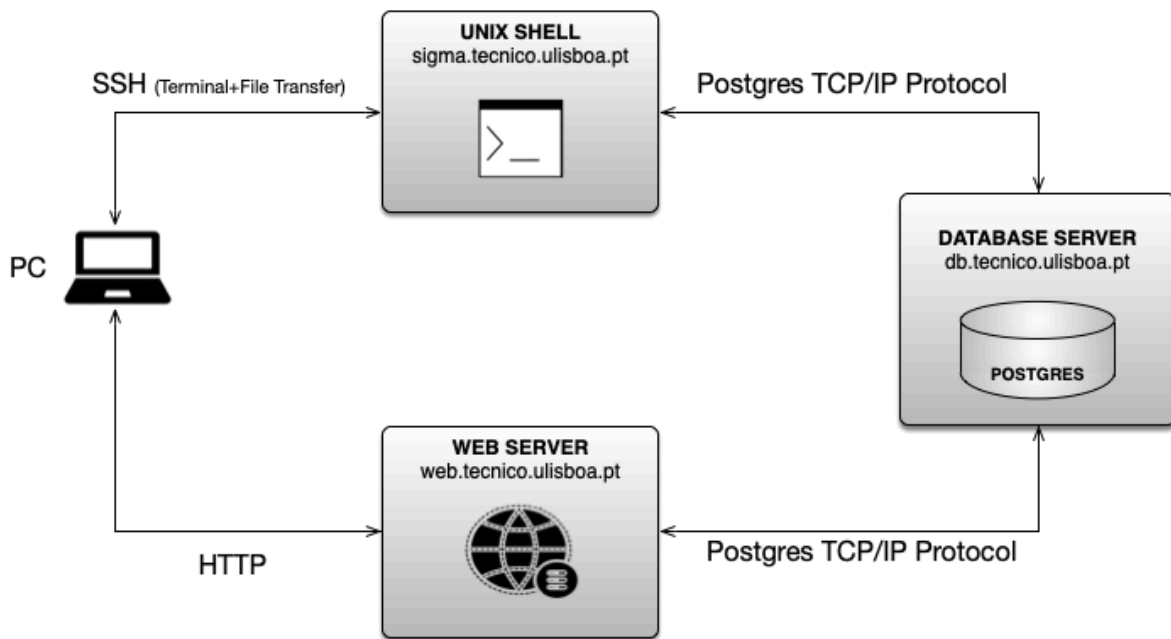


Bases de Dados

Lab 1: Introdução ao ambiente

O Sistema de Gestão de Base de Dados PostgreSQL bem como os restantes serviços de Shell Unix, Sistema de servidor Web (onde é executado código Python) encontram-se disponíveis na infraestrutura do IST, representada na Figura 1:



Os servidores sigma.tecnico.ulisboa.pt e web.tecnico.ulisboa.pt partilham os sistemas de ficheiros, que são editados no sigma.tecnico.ulisboa.pt e visualizados na WWW a partir do web2.tecnico.ulisboa.pt¹.

Nota: Previamente à realização do laboratório deve rever os conceitos [Shell Unix](#)², [SSH](#)³ e [SCP](#)⁴

¹ Atenção: usaremos o **web2.tecnico.ulisboa.pt** porque apenas este corre Python3; em web.tecnico.ulisboa.pt está instalada a versão 2 do Python.

² <https://www.learnenough.com/command-line-tutorial/basics>

³ [https://en.wikipedia.org/wiki/Ssh_\(Secure_Shell\)](https://en.wikipedia.org/wiki/Ssh_(Secure_Shell))

⁴ https://en.wikipedia.org/wiki/Secure_copy_protocol

Configurações e acesso ao PostgreSQL

Para começar a usar o sistema **PostgreSQL** disponível na infraestrutura do IST, deverá proceder aos seguintes passos:

1. Para poder fazer o que é solicitado neste laboratório, deverá aceder primeiro à página de self-service da DSI:

<https://selfservice.dsi.tecnico.ulisboa.pt/>

e ativar os serviços shell, web e cgi

2. Para aceder à infraestrutura Cloud do IST tem de usar um programa designado genericamente por “Secure Shell” que permite estabelecer uma sessão segura com o *host*/anfitrião **sigma.tecnico.ulisboa.pt**.

- a. **Windows 10, Linux ou Mac OS:** Deve abrir um terminal usar o comando⁵

```
ssh istxxxx@sigma.tecnico.ulisboa.pt
```

- b. **Windows 8.1 e anteriores:** Pode usar um dos programas de “SSH Secure Shell” ou “[PuTTY](#)” (que deve fazer download) e ligar-se ao anfitrião/host **sigma.tecnico.ulisboa.pt**

Nota: A autenticação no **sigma** é sempre efetuada através as suas credenciais do Fénix (nome de utilizador e password do sistema Fénix) para entrar na sua conta do sigma. Por razões de segurança, ao digitar a password, **os caracteres não aparecem** no écran.

Nota: Na primeira vez que estabelece uma ligação segura com um novo servidor (host) ser-lhe-á solicitado que aceite a chave ssh. Pode responder afirmativamente. Na maior parte dos casos, isto corresponde a escrever ‘yes’ ou a pressionar o botão ‘Ok’.

3. Ao entrar em sessão no **sigma** deverá ver uma linha de comandos de Shell Unix com um *prompt* semelhante a:

```
ist12345@sigma01:~$
```

4. Para obter a password da sua conta no **Postgres** no sigma, execute o comando:

```
psql_reset
```

5. Para entrar em sessão em sessão no **Postgres** use o comando:

⁵ Caso obtenha o erro **ssh: connect to host sigma.tecnico.ulisboa.pt port 22: Connection timed out**, verifique se a sua firewall não está a bloquear ligações SSH

```
psql -h db.tecnico.ulisboa.pt -U istxxx
```

Onde **istxxxxx** deve ser substituído pelo seu nome de utilizador no sistema Fénix, e deverá entrar com a password obtida no comando anterior. Deve digitar o comando com os espaços cf. indicado.

6. Irá obter uma sessão para o Postgres que será indicada pela alteração do prompt da linha de comando para:

```
ist24950=>
```

7. Se quiser atualizar a password de acesso ao **Postgres** pode usar o comando:

```
ALTER USER istxxxx WITH PASSWORD 'mypassxxxx';
```

em que **mypassxxxx** corresponde à nova password de acesso ao Postgres.

8. Use o comando `\q` para sair do sistema e voltar ao terminal.
9. Repare que a prompt da linha de comandos voltou a ser:

```
ist12345@sigma01:~$
```

Transferência de ficheiros para o Sigma

10. Faça download do ficheiro Zip que acompanha este laboratório
11. Para copiar ficheiros do seu PC para a sua área de ficheiros do sigma.tecnico.ulisboa.pt deverá usar um programa específico de cópia segura (Secure Copy).

- **Windows 10, Linux ou Mac OS:** Abrir um terminal e usar o comando “scp”:

```
scp <file> istxxxx@sigma.tecnico.ulisboa.pt:<path-to-folder>
```

Nota: Caso esteja a fazer o download a partir de um dos PCs dos laboratórios em Linux, então esta cópia não será necessária dado que o download do browser já deixa os ficheiros na vossa área no sigma.

Nota: Opcionalmente pode instalar o “[FileZilla](https://filezilla-project.org)”⁶ para fazer a transferência dos ficheiros mais facilmente. Este software funciona em Windows, Linux e Mac OS.

12. Copie o ficheiro *bank.sql* para a diretoria raiz (~) da sua área no **sigma**.

13. Copie o ficheiro *test.cgi* para dentro pasta ~/web da sua área no **sigma**.

Utilização da linha de comando

Em geral todos os sistemas de gestão de base de dados dispõem de uma interface de linha de comando através da qual é possível executar instruções SQL e também outros comandos de administração e manutenção do sistema.

Nota: Todos os comandos de administração começam com \ (contra-barra ou barra invertida). Todos os restantes comandos começam com uma instrução SQL (**select**, **insert**, **update**, **delete**, **create**, ...) e terminam com ; (ponto e vírgula).

14. Introduza novamente na sua sessão Postgres tal como explicado no passo 5.

15. Uma vez dentro do sistema, utilize o comando \h [?] para obter informação sobre todos os comandos SQL disponíveis. Use \q [?] para sair.

16. Utilize o comando \? [?] para obter informação sobre todos os comandos de administração disponíveis.

17. Se precisar de sair do programa, pode utilizar a qualquer momento o comando: \q [?]

Criação da base de dados exemplo ‘Bank’

O ficheiro *bank.sql* contém um conjunto de instruções SQL para criar a base de dados de exemplo ilustrada na Figura 2.

Para criar a base de dados é necessário criar as tabelas e carregar os registos de cada tabela. A criação das tabelas é feita com recurso à instrução **CREATE TABLE**.

⁶ <https://filezilla-project.org>

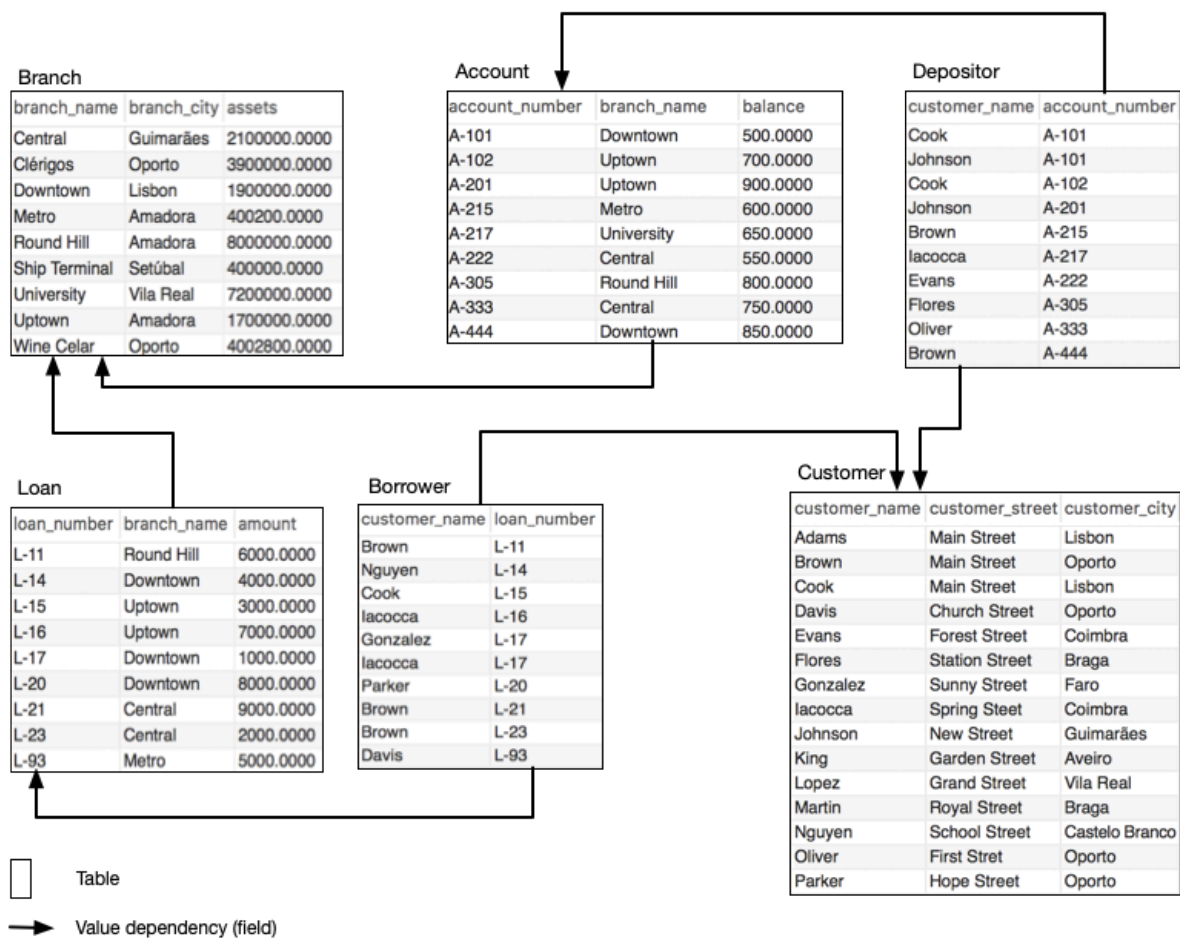


Figura 2. Organização das tabelas da base de dados de exemplo

Por exemplo, a tabela de clientes pode ser criada com a seguinte instrução:

```
create table customer
(
  customer_name      varchar(80) not null unique,
  customer_street    varchar(255)not null,
  customer_city      varchar(30) not null,
  constraint pk_customer primary key(customer_name));
```

Esta instrução especifica o nome da tabela, os nomes das três colunas, o tipo de cada coluna, e ainda restrições tais como os valores não poderem ser **NULL** e o facto da chave primária da tabela ser o nome do cliente.

Os registos de cada tabela são carregados através de instruções do tipo **INSERT**. Por exemplo:

```
insert into customer values ( 'Luis' , 'Rua do Cima' , 'Musgueira' );
```

Nesta instrução são especificados, respetivamente, os valores de cada coluna pela mesma ordem em que estes foram definidos aquando da criação da tabela⁷. Esta instrução resulta na criação de um novo registo na tabela de clientes.

Note-se que o ficheiro *bank.sql* inclui instruções para inserir mais registos na base de dados do que aqueles que se encontram exemplificados na Figura 2. Estes registos serão usados para realizar vários testes sobre a base de dados. Em aulas de laboratório futuras, iremos utilizar esta base de dados para demonstrar vários dos conceitos da disciplina.

Nota: Muito embora também possa ser utilizada a interface gráfica **pgAdmin**⁸, como forma de interagir com o sistema de gestão de bases de dados **Postgres**, nas aulas de laboratório iremos sobretudo utilizar a interface de linha de comandos.

18. Entre na sua área do sigma, como explicado no passo 2 (caso seja necessário, navegue até ao diretório onde guardou o ficheiro *bank.sql*).
19. Entre no Postgres, como explicado nos passos 4—6.
20. Execute o comando que irá executar todos os comandos para criar a base de dados de exemplo do Bank:

```
\i bank.sql
```

Para carregar e executar as instruções SQL que estão no ficheiro *bank.sql*.

O Postgres produz algumas mensagens à medida que executa as instruções do ficheiro. No final da execução, a base de dados de exemplo está criada.

21. Para listar as tabelas da base de dados, use o comando: `\d`
22. O tempo que o sistema demora a responder a algumas consultas é um fator importante quando o volume de dados é considerável. Execute o comando:

⁷ Existem outras variantes da instrução **INSERT** com as quais é possível especificar os valores por outra ordem, ou especificar apenas alguns dos valores pretendidos deixando os restantes a **NULL** ou com o valor por omissão (*default value*, que neste caso não foi especificado)

⁸ A interface **pgAdmin** encontra-se disponível em <https://www.pgadmin.org>

`\timing` para ativar e desativar a cronometragem do tempo de execução dos comandos SQL.

23. Uma vez dentro da sessão Postgres, poderá efetuar algumas consultas introduzindo comandos SQL, nomeadamente:

- ver a lista completa de clientes: `SELECT * FROM customer;`
- ver a lista completa de contas: `SELECT * FROM account;`
- ver o saldo da conta A-101:

`SELECT balance FROM account WHERE account_number='A-101';`

- ver todos os clientes que não são depositantes (i.e. não têm contas):

```
SELECT * FROM customer
WHERE customer_name NOT IN (
  SELECT customer_name FROM depositor);
```

Em aulas futuras irá aprender a obter a resposta a consultas mais complexas.

Obter informação sobre o esquema de uma base de dados

Para uma base de dados existente no sistema mas sobre a qual não haja documentação, é possível usar instruções especiais para obter informação sobre as respetivas tabelas. Geralmente estes mecanismos são proprietários e diferentes para cada sistema. No Postgres estas funcionalidades estão disponíveis através do comando `\d` e variantes.

24. Utilize o comando `\l` ('L' minúsculo) para obter informação de todas as bases de dados existentes no sistema.

25. Obtenha informação sobre as tabelas de clientes e contas:

`\d customer`

`\d account`

Confirme que a estrutura dessas tabelas está de acordo com as instruções que foram dadas no ficheiro bank.sql.

26. Use o comando `\q` para sair do sistema e voltar ao terminal.

Resumo dos comandos Postgres utilizados

<code>\h</code>	Obter informação sobre todos os comandos SQL disponíveis.
<code>\q</code>	Sair da linha de comando do Postgres.
<code>\?</code>	Obter informação sobre todos os comandos de administração.
<code>\i file</code>	Executar os comandos contidos no ficheiro indicado.
<code>\c database</code>	Ligar à base de dados indicada.
<code>\d</code>	Listas as tabelas da base de dados atual.
<code>\timing</code>	Mostrar o tempo de execução das consultas.
<code>\l</code>	Listar todas as bases de dados existentes no sistema.
<code>\d table</code>	Obter informação sobre a estrutura de uma tabela.

Desenvolvimento de aplicações com bases de dados Postgres : Exemplo em Python

Os passos seguintes têm por objetivo testar a ligação à base de dados através de um script Python.

27. Certifique-se que tem instalado um editor de texto apropriado, como o Sublime Text⁹ ou o Visual Studio Code¹⁰.
28. Edite o ficheiro `test.cgi` e coloque o seu nome de utilizador e password (dada pelo `psql_reset`) nas variáveis `DB_USER` e `DB_PASSWORD`, respetivamente.
29. Usando um cliente de SCP (“WinSCP” em Windows; “scp” em Linux) coloque o ficheiro `test.cgi` dentro da pasta `web` da sua área no cluster `sigma.tecnico.ulisboa.pt`

```
scp test.cgi istxxxx@sigma.tecnico.ulisboa.pt:~/web/test.cgi
```

30. Na sua área no sigma, defina as permissões do ficheiro `test.cgi` como executável.

```
chmod 755 test.cgi
```

Nota: O ficheiro.cgi não deve ter permissões de escrita para o owner nem nenhum dos diretórios onde se encontra. `chmod -R 755 ~/web` aplica recursivamente as permissões à directoria `web`.

31. Abra o browser e aceda ao URL:

`http://web2.tecnico.ulisboa.pt/istxxxxx/test.cgi`

onde **`istxxxxx`** é o seu nome de utilizador no sistema Fénix.

32. Confirme que a página é gerada sem erros.

Nota: Caso a página não esteja a ser gerado uma vez introduzido o endereço, isso deve-se provavelmente ao facto do *username* ou da *password* introduzidas do ficheiro `test.cgi` não estar(em) correta(s).

⁹ <https://www.sublimetext.com>

¹⁰ <https://code.visualstudio.com>

Nota: Caso seja necessário instale o driver de PostgreSQL do python3 usando o comando:

pip3 install psycopg2

33. Abra o ficheiro `test.cgi` num editor e localize as chamadas às seguintes funções:

- `connect(...)`
- `queryexecute(...)`
- `cursor(...)`
- `close(...)`

Esta é tipicamente a sequência de chamadas para interagir com uma base de dados a partir de Python.

34. No ficheiro `test.cgi`, localize a consulta à base de dados:

SELECT * FROM account;

Compare o resultado desta consulta, que aparece no browser, com o resultado que obteve através da linha de comando.

35. No ficheiro `test.cgi`, localize o sítio onde está a ser construída uma tabela HTML para mostrar os resultados:

- `<table>` e `</table>` definem o início e fim da tabela HTML
- `<thead>` `</thead>` definem o início e fim do cabeçalho da tabela HTML
- `<tbody>` `</tbody>` definem o início e fim do corpo da tabela HTML
- `<tr>` e `</tr>` definem o início e fim de uma linha da tabela HTML
- `<td>` e `</td>` definem o início e fim de uma célula na tabela HTML
- Esta é a forma típica de construção de tabelas em HTML. Repare que a tabela está a ser construída dentro de um ciclo que percorre os resultados de `cursor.execute(query)`

36. No browser, abra o código fonte da página e compare esse código HTML com o código Python do ficheiro de script `test.cgi`.

Utilização do pgAdmin

37. Instale a versão mais recente do **pgAdmin**¹¹ (ou a versão 3) que lhe permitirá aceder diretamente ao servidor de base de dados Postgres.
38. Em Linux, Windows ou Mac OS abra a aplicação **pgAdmin**.
39. Crie uma ligação para o host `db.tecnico.ulisboa.pt`, port 5432 (assegure-se de que o seu *username* e a sua *password* estão inseridas corretamente).
40. Expanda a árvore de ligações até encontrar a ligação à base de dados correspondente ao seu utilizador **istxxxx**.
41. Abra o nó *Databases* e depois o nó *Schemas* e depois o nó `public>Tables` onde poderá visualizar as tabelas da base de dados exemplo que criou anteriormente.
42. Abra uma janela de SQL utilizando a sua ligação (em “Tools > Query Tool”) e execute uma query para obter a lista completa de clientes: `SELECT * FROM customer;`

¹¹ A interface pgAdmin encontra-se disponível em <https://www.pgadmin.org>

Apêndice: Comandos Unix

BASIC SHELL	
TAB	Complete
Up / Down	Command History
CTRL + L	Clear
CTRL + S	Stop scroll (freeze terminal)
CTRL + Q	Continue scroll (unfreeze terminal)
MOVEMENT	
CTRL + A	Beginning of line (Home)
CTRL + E	End of line (End)
ESC - B / CTRL + LEFT	Skip word left (Backward)
ESC - F / CTRL + RIGHT	Skip work right (Forward)
CTRL + X + X	Toggle current position with start of line
CTRL + _	Undo
DELETE/CUT/PASTE	
CTRL + U	Kill line
CTRL + K	Kill to end of line
CTRL + W	Kill word backwards
CTRL + Y	Paste (Yank)
TEXT	
ALT + U	Convert next word to upper case
ALT + L	Convert next work to lower case
PROCESS CONTROL	
CTRL + C	Break
CTRL + D	Close / Exit
CTRL + Z	Stop