



SISTEMA DISTRIBUÍDO DE RESERVA E COMPRA DE BILHETES

TRABALHO PRÁTICO – META 1

PROGRAMAÇÃO DISTRIBUÍDA

LEI- DA 2022/2023

| | |
|-----------------|------------|
| Eduardo Bento | 2020139115 |
| Henrique Negrão | 2018012326 |
| Rúben Santos | 2019116244 |

OBJETIVOS DO TRABALHO

- Criação de um Sistema Distribuído de Reserva e Compra de Bilhetes;
- Escrito em Java;
- Composto por:
 - **Programas Cliente** – aplicação através da qual os utilizadores interagem com o Sistema;
 - **Programas Servidor** – aplicação responsável por toda a lógica de negócio;



OBJETIVOS DO TRABALHO

CLIENTE



O cliente pode adquirir bilhetes para espetáculos. Para isto utilizará uma interface.

É iniciado fornecendo-lhe o endereço IP e o porto de escuta de um servidor.

Cada vez que se dá uma mudança na base de dados do servidor a sua vista é atualizada.

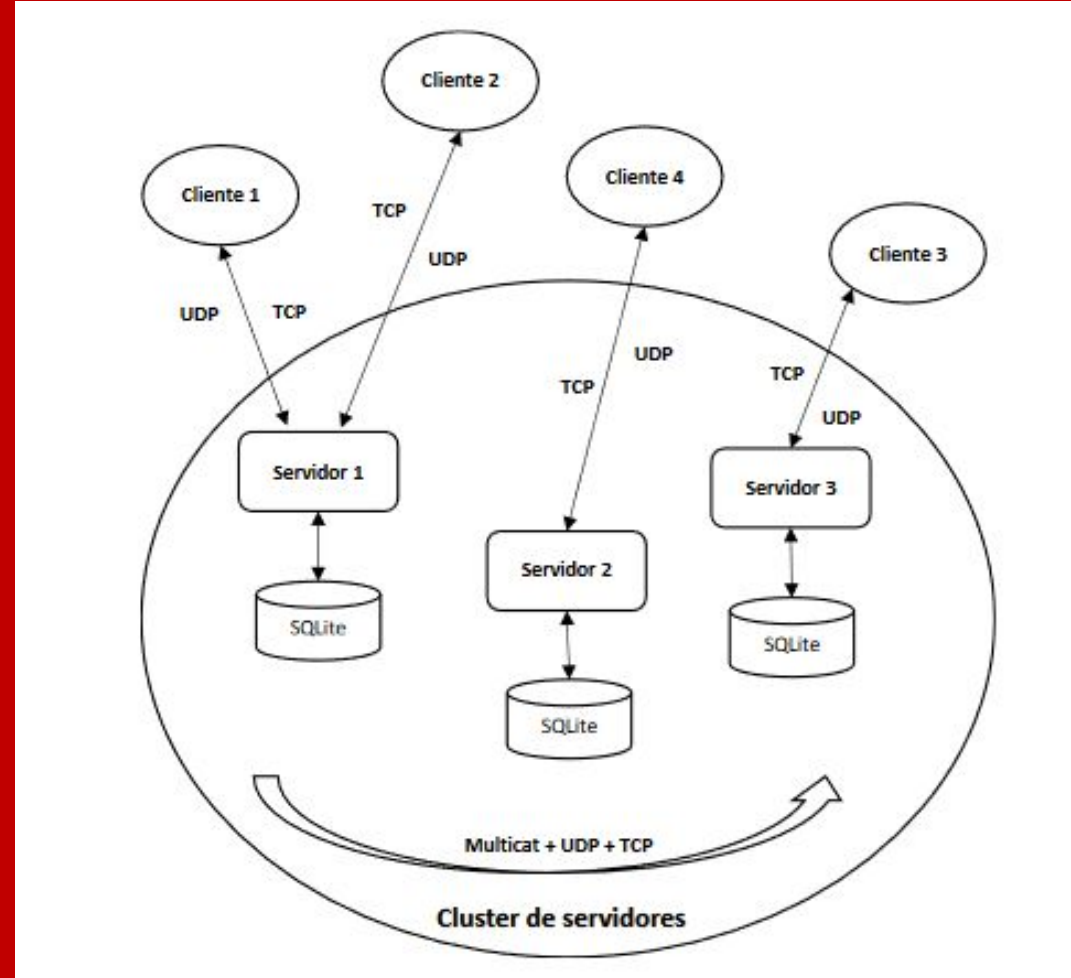
Podem existir vários servidores, cada um com o seu porto de escuta. São eles que fazem os pedidos à base de dados.

Interagem entre si, atualizando as suas bases de dados de modo a estarem todas iguais.



SERVIDORES

MODELO FÍSICO DA BASE DE DADOS



FUNCIONALIDADES IMPLEMENTADAS

| Atualização das réplicas da BD | Feito ? |
|---|---------|
| <p>Mecanismos de obtenção de informação por parte dos clientes:</p> <ul style="list-style-type: none">-> (1) cliente envia pedido ao servidor via TCP;-> (2) servidor submete query à BD local se não estiver envolvido numa operação de atualização da BD (caso contrário devolve indicação de insucesso ao cliente);-> (3) servidor devolve resultado ao cliente via TCP; | ✓ |
| <p>Mecanismo indicado no enunciado quando é feita uma alteração à BD por parte de um servidor:</p> <ul style="list-style-type: none">-> (1) O cliente solicita uma operação que envolve alteração da BD ao servidor via TCP;-> (2) Se não estiver envolvido noutra operação que envolva acesso à BD (e.g., PREPARE enviado ou recebido, outro pedido de um cliente local a ser processado), o servidor envia um PREPARE. Caso contrário, devolve ao cliente uma indicação de insucesso;-> (3) O servidor aguarda um ACK de todos os servidores supostamente ativos. Se faltar algum ACK, volta a enviar o ACK uma segunda vez. <p>Se voltar a acontecer, cancela a operação e devolve ao cliente uma indicação de insucesso;</p> <ul style="list-style-type: none">-> (4) Se recebeu os ACK esperados, o servidor envia um COMMIT, atualiza a sua BD local, devolve ao cliente uma indicação de sucesso e notifica todos os clientes locais (em princípio, autenticados) para que as suas vistas sejam atualizadas de forma assíncrona; <p>Nota: devem existir mecanismos de timeout de espera de respostas nos servidores.</p> | ✓ |

FUNCIONALIDADES IMPLEMENTADAS

| Arquitetura / concorrência / canais de comunicação | Feito ? |
|--|---------|
| Estrutura das mensagens trocadas (texto, objetos serializados de tipos distintos, conteúdo dos heartbeats, etc.) | ✓ |
| Servidor: thread dedicada a receber datagramas UDP de contacto de clientes | ✓ |
| Servidor: thread dedicada a receber datagramas UDP no grupo de multicast (heartbeats, Prepare, Commit, etc.) | ✓ |
| Servidor: thread dedicada ao envio periódico de heartbeats ou esquema alternativo (e.g., Timer) | ✓ |
| Servidor: Thread dedicada à receção de pedidos de ligação pelos clientes via TCP | ✓ |
| Servidor: thread para comunicação com cada cliente ligado via TCP | ✓ |
| Cliente: thread nos clientes dedicada à receção de notificações assíncronas (TCP) | ✓ |
| "Servidor: ligações TCP temporárias (e threads adicionais associadas) para transferência de BD (ficheiro ou outro tipo de informação) se foresta a solução adotada." | ✓ |

FUNCIONALIDADES IMPLEMENTADAS

| Funcionalidades | Feito ? |
|---|---------|
| Servidor: aprende a existência dos restantes de forma dinâmica (heartbeats via multicast) e faz a gestão adequada da lista de servidores ativos (ordena em função da carga e esquece os que não dão sinais de vida há pelo menos 35 segundos) | ✓ |
| Cliente: (1) contacto de um servidor via UDP; (2) receção de um lista de servidores ativos; e (3) ligação TCP ao servidor com menos carga (se falhar o primeiro, tenta o segundo e assim sucessivamente) | ✓ |
| Utilizador: registo (nome + username + password) | ✓ |
| Utilizador: autenticação (nome + username) | ✓ |
| Utilizador: edição dos dados de registo | ✓ |
| Administrador: inserção de um espetáculo através de um ficheiro (qualquer erro no ficheiro cancela a operação) | ✓ |
| Administrador: tornar um espetáculo visível | ✓ |
| Administrador: eliminação de um espetáculo sem reservas pagas associadas. As reservas não pagas também são eliminadas da BD. | ✓ |
| Utilizador: consulta de espetáculos (existência de filtros) | ✓ |
| Utilizador: visualização dos lugares disponíveis de um espetáculo que irá decorrer, no mínimo, dentro de 24 horas | ✓ |

FUNCIONALIDADES IMPLEMENTADAS

| Funcionalidades | Feito ? |
|--|---------|
| Utilizador: seleção dos lugares (operação local à aplicação Cliente) | ✓ |
| Utilizador: submissão da reserva | ✓ |
| Utilizador: consulta de reservas que aguardam pagamento | ✓ |
| Utilizador: eliminação de uma reserva não paga | ✓ |
| Utilizador: pagamento da reserva | ✓ |
| Utilizador: consulta de reservas pagas | ✓ |
| "Servidor: criação de uma BD vazia (número de versão 1) no arranque quando não existe localmente e não ouve qualquer heartbeat. Não envia heartbeats durante esta fase." | ✓ |
| "Servidor: obtenção de uma BD no arranque quando não existe localmente e ouve pelo menos um heartbeat. Não envia heartbeats durante esta fase." | ✓ |
| "Servidor: obtenção de uma BD quando existe localmente, mas ouve pelo menos um heartbeat com versão superior. Deve fornecer uma lista atualizada de servidores aos clientes, onde não aparece, e fechar as respetivas ligações TCP. Também deveria passar a enviar heartbeats onde indica que não está disponível durante a obtenção da BD (não aceita clientes)." | ✓ |

FUNCIONALIDADES IMPLEMENTADAS

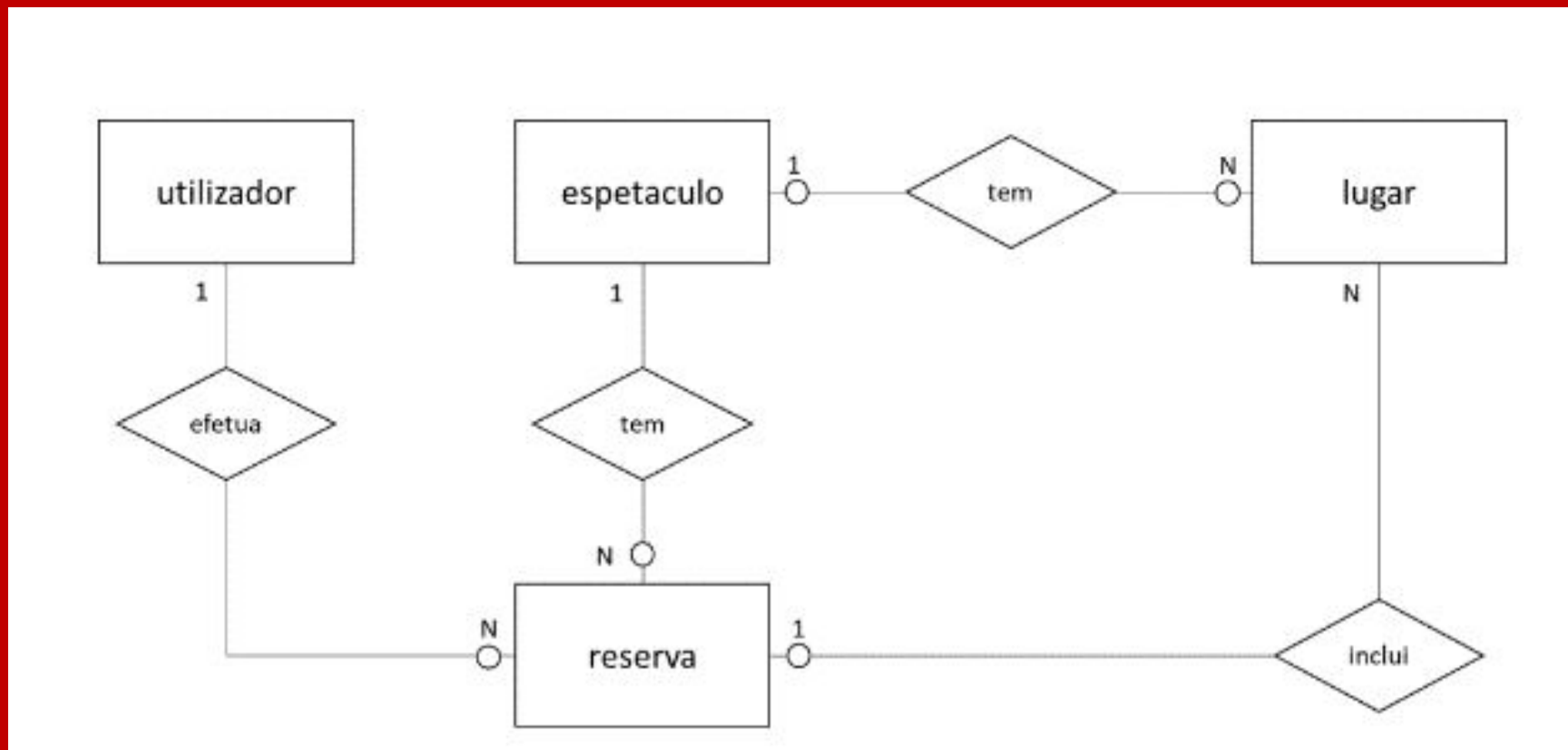
| Funcionalidades | Feito ? |
|---|---------|
| "Cliente: quando perde a ligação TCP a um servidor, deve tentar ligar-se a outro constante da lista que detém. Deveria ser pouco perceptível para o utilizador." | ✓ |
| Base de dados replicadas consistentes: sempre que existe uma alteração na BD, verificar se esta se propaga a todas as réplicas. | ✓ |
| "Cliente: atualização assíncrona da vista / notificações assíncronas na vista (e.g., lista de espetáculos se estiver a ser visualizada, lugares disponíveis num espetáculo que esteja a ser visualizado)" | ✓ |



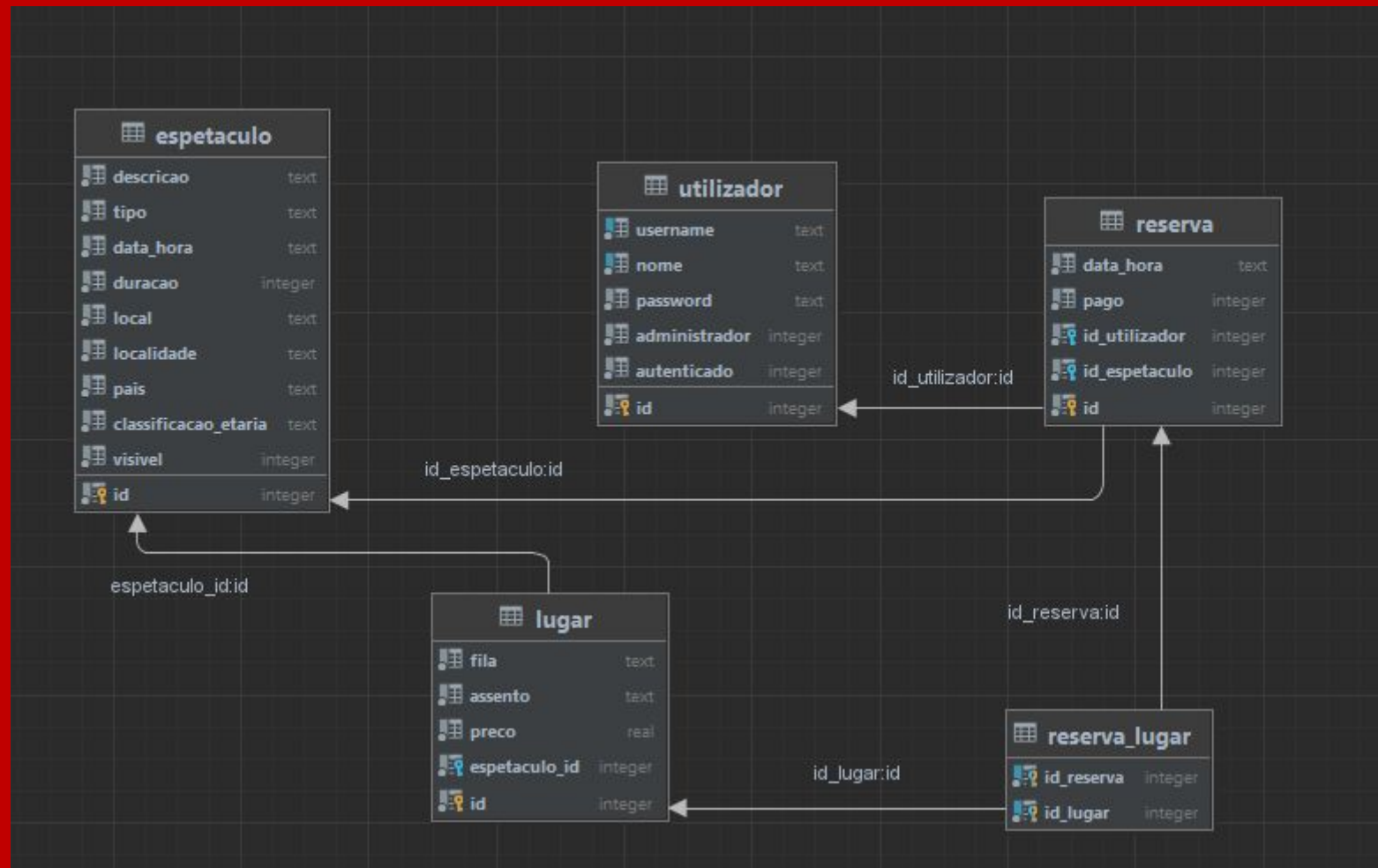
FUNCIONALIDADES IMPLEMENTADAS

| Não especificadas no enunciado | Feito ? |
|---|---------|
| Cliente: Caso o espetáculo seja eliminado pelo administrador, e o cliente estivesse a escolher um lugar desse espetáculo, é retirado da vista de modo a não conseguir comprar bilhetes para esse espetáculo | ✓ |
| Servidor: De modo a criar uma melhor consistência nas operações de update da BD e do ponto de vista do utilizador (não ter de fazer as coisas de novo) criámos uma queue de espera para o caso da operação não ser possível | ✓ |
| | |
| | |
| | |
| | |
| | |
| | |

MODELO ER DA BASE DE DADOS



MODELO FÍSICO DA BASE DE DADOS



COMUNICAÇÃO COM A BASE DE DADOS

```
public class DBHandler {  
    52 usages  
    private final Log LOG = Log.getLogger(DBHandler.class);  
    22 usages  
    private Connection connection;  
  
    1 usage  🧑 eduardo +1  
    public DBHandler(String path) {  
        try {  
            connection = DriverManager.getConnection( url: "jdbc:sqlite:" + path);  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

THREADS DO SERVIDOR

HeartBeat:

- Envio
- Remover
- Receção

Ping UDP - recebe pings do clientes e envia a lista de servidores

ClientManagment - cria um comunicação tcp quando um cliente de liga ao serverSocket

ClientReceiveMessage - Recebe pedidos do cliente

QueueUpdate - Quando existe uma atualização a decorre o pedido do cliente é colocado em lista de espera (verifica a lista de pedidos a cada 500ms).

THREADS DO CLIENTE

- Thread de resposta - onde recebe todas as mensagens enviadas pelo servidor (assíncronas/não assíncronas)

COMUNICAÇÃO ENTRE OS DOIS

Para a comunicação utilizamos um objeto serializado ClientData que contém um Enumerador (que corresponde ao tipo de ação a executar) e um objeto com a informação necessária para o servidor fazer o pedido à base de dados.

Exemplo:

- Cliente faz um request de para um registo de um utilizador
- A ação vai ser do tipo ClientAction.REGISTER, e objeto enviado vai ser um Objeto do tipo triple<Username,name,password>
- O servidor recebe este objeto verifica o tipo de ação e executa a função respectiva a base de dados não atualizando de imediato a base de dados local, guarda numa lista os comandos necessários e só depois de verificar que todos os servidores estão online é que atualiza a sua db



MANUAL DO UTILIZADOR

MENU INICIAL



LOGIN

Ao iniciar o programa o utilizador pode começar por autenticar-se

Para fazer isto basta introduzir:

- o seu username
- a sua password

ATENÇÃO: Qualquer informação importante irá aparecer na parte de baixo da interface. Esteja atento!

The screenshot shows a web application interface with a red header. In the center of the header is a Christmas tree icon inside a square frame. Below the frame are five black buttons with white letters: S, D, R, C, and B. The main content area is divided into two sections. The top section is titled 'Login' and contains the text 'Faça login para continuar'. It has two input fields: 'Username' and 'Password'. Below the fields is a black button labeled 'Conectar'. At the bottom of the section is a link: '[Registe-se aqui!](#)'. The bottom section is titled 'Criar conta' and contains three input fields: 'Username', 'Nome', and 'Password'. Below the fields is a black button labeled 'Registrar'. At the bottom of the section is a link: '[Já registrado? Conecte-se](#)'.



REGISTO

No entanto, caso ainda não tenha dados de acesso, pode registar-se

Para fazer isso basta fornecer:

- o seu username pretendido
 - o seu nome
- a sua password pretendida

ATENÇÃO: Dois utilizadores não podem ter o mesmo username e/ou nome

MENU CLIENTE



ESCOLHER

Caso a autenticação tenha sido efetuada com sucesso o utilizador pode então escolher entre as funcionalidades:

- Editar Dados Pessoais -> pode editar os seus dados (username, nome, password)
- Adquirir Bilhetes -> pode seleccionar um espetáculo (que vá ocorrer depois 24 horas depois das horas atuais) e adquirir bilhetes para esse espetáculo



Editar Dados Pessoais



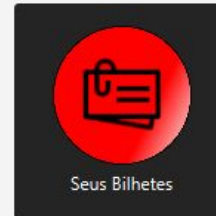
Adquirir Bilhetes



Consultar Espetáculos



Reservas não Pagas



Seus Bilhetes



ESCOLHER

- Consultar Espetáculos -> pode procurar espetáculos com base em filtros
- Reservas não pagas -> pode visualizar e pagar as suas reservas não pagas num tempo limite de 10 segundos
- Seus Bilhetes -> pode visualizar os seus bilhetes já adquiridos

EDITAR DADOS PESSOAIS



EDITAR

Neste menu o utilizador pode editar:

- o seu username
- o seu nome
- a sua password

[Voltar Atrás](#)

Editar

Username ▼

Mudar



EDITAR

Apenas selecione o que quer editar,
escreva para o que quer mudar
na caixa de texto em baixo
e clique no botão!

CONSULTAR ESPETÁCULOS



FILTROS

Neste menu o utilizador pode consultar os espetáculos (visíveis) com base em filtros selecionáveis

Por exemplo:

- Um espetáculo em Coimbra
- Um espetáculo dia 26 de dezembro
- Um espetáculo em Portugal

[Voltar Atrás](#)

| | |
|--|---|
| <input type="checkbox"/> descricao | <input type="text" value="Insira uma descrição"/> |
| <input type="checkbox"/> tipo | <input type="text" value="Insira uma tipo"/> |
| <input checked="" type="checkbox"/> data_hora | <input type="text" value="26/12/2022"/> |
| <input type="checkbox"/> duracao | <input type="text" value="Defina uma duração"/> |
| <input type="checkbox"/> local | <input type="text" value="Escolha um local"/> |
| <input checked="" type="checkbox"/> localidade | <input type="text" value="Coimbra"/> |
| <input type="checkbox"/> classificacao_etaria | <input type="text" value="Escolha a classificaçã"/> |
| <input checked="" type="checkbox"/> país | <input type="text" value="Portugal"/> |

Descrição: Teste 5
Tipo: Música
Data: 2022-12-26 22:30:00
Duração: 75 minutos
Local: Convento São Francisco –Sala D. Afonso Henriques
Localidade: Coimbra
País: Portugal
Classificação Etária: 6



FILTROS

ADQUIRIR BILHETES



SELEÇÃO

Neste menu o utilizador começa por selecionar um espetáculo que irá decorrer 24 horas após as horas atuais

Após selecionar irá então entrar num menu de seleção de lugares

Neste menu o utilizador pode selecionar o(s) lugar(es) que pretende reservar

[Voltar Atrás](#)

Descrição: Teste 5
Tipo: Música
Data: 2022-12-26 22:30:00
Duração: 75 minutos
Local: Convento São Francisco –Sala D. Afonso Henriques
Localidade: Coimbra
País: Portugal
Classificação Etária: 6

[Voltar Atrás](#)

| | | | | | | | | | | |
|----|--|--|--|--|--|--|--|--|--|--|
| A: | | | | | | | | | | |
| B: | | | | | | | | | | |
| C: | | | | | | | | | | |
| D: | | | | | | | | | | |
| E: | | | | | | | | | | |
| F: | | | | | | | | | | |
| G: | | | | | | | | | | |
| H: | | | | | | | | | | |
| I: | | | | | | | | | | |

[Reservar](#)



LUGARES

Os lugares selecionados aparecem a verde e os não disponíveis a vermelho (e nem são possíveis de selecionar)

Caso no tempo em que está a selecionar o lugar outro utilizador reserve esse(s) lugar(s), o utilizador será informado

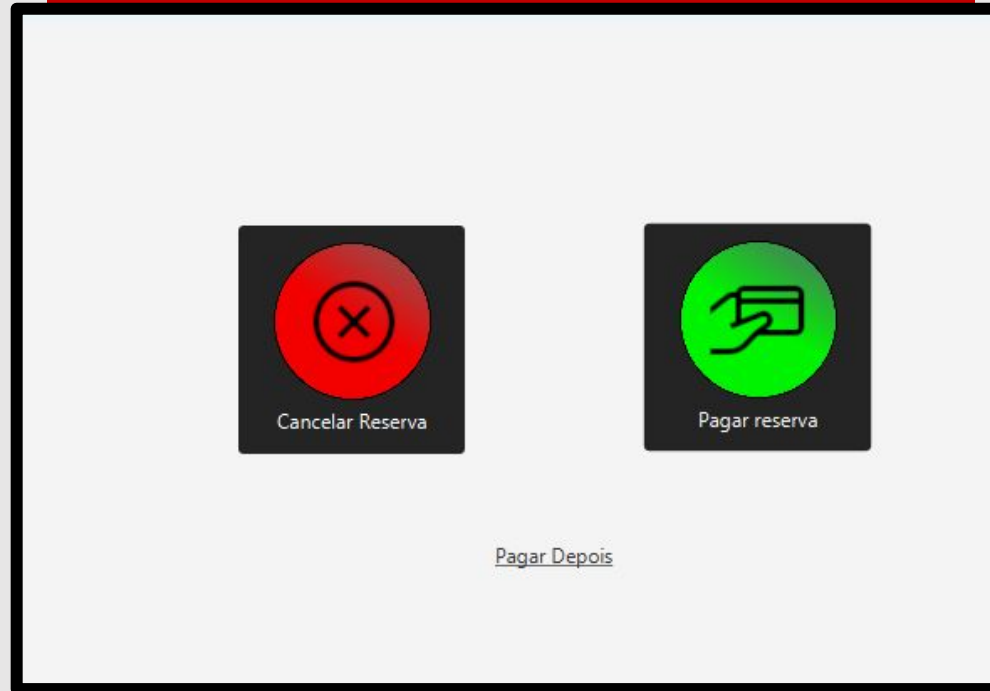
PAGAMENTO



OPÇÕES

Após clicar no botão de reservar pode escolher entre:

- Cancelar Reserva -> cancela a reserva que fez e mete os lugares de novo disponíveis
- Pagar Reserva -> obtem os bilhetes dos lugares pretendidos



OPÇÕES

- Pagar Depois— volta ao menu inicial, podendo pagar mais tarde os bilhetes num intervalo de 10 segundos

ATENÇÃO: Poderá pagar no menu “Reservas não Pagas”

RESERVAS NÃO PAGAS



PAGAR

No caso de o utilizador ter escolhido “Pagar Depois” após ter feito a sua reserva será possível pagar neste menu

No entanto, deve fazê-lo num intervalo de 10 segundos a começar desde o momento em que submeteu a sua reserva não paga

[Voltar Atrás](#)

Reservas não pagas

Espetáculo: 1 - 2022-12-26 22:30:00
Data: 2022-12-04 07:11:57
Lugar: F5 - 10.0 €

| |
|--|
| |
| |
| |
| |
| |
| |

EliminarPagar



CANCELAR

Caso queira também pode eliminar/cancelar a reserva

ATENÇÃO: Caso não pague, a reserva irá ser revertida e os lugares irão passar a estar disponíveis de novo

SEUS BILHETES



VER

Neste menu o utilizador pode ver os bilhetes que adquiriu.

[Voltar Atrás](#)

Reservas efetuadas

Espetáculo: 1 2022-12-26 22:30:00
Data: 2022-12-04 07:11:12
Lugar: E5
Lugar: E6

| |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |



VER

Vê a informação dos lugares da sua reserva paga

MENU ADMINISTRADOR



ESCOLHER

Dados de acesso:

- username: admin
- password: admin

Caso a autenticação tenha sido efetuada com sucesso o administrador pode então escolher entre as funcionalidades:

- Editar Dados Pessoais -> pode editar os seus dados (username, nome, password)



Editar Dados Pessoais



Gerir Espetáculos



Consultar Espetáculos



ESCOLHER

- Gerir Espetáculos -> pode procurar espetáculos com base em filtros
- Consultar Espetáculos -> pode procurar espetáculos com base em filtros

GERIR ESPETÁCULOS



GERIR

Neste menu o administrador pode:

- Adicionar um espetáculo -> escreve na caixa de texto o nome do ficheiro de texto onde se encontram as informações do espetáculo a adicionar e clica no botão

ATENÇÃO: Caso surja algum erro por causa de informação não válida no ficheiro de texto, será informado sobre o mesmo

[Voltar Atrás](#)

Especáculos

Adicionar

Remover

Gerir visibilidade

Descrição: Teste 5
Tipo: Música
Data: 2022-12-26 22:30:00
Duração: 75 minutos
Local: Convento São Francisco - Sala D. Afonso Henriques
Localidade: Coimbra
País: Portugal
Classificação Etária: 6
Visível: sim



GERIR

- Remover um espetáculo -> seleciona o espetáculo e clica no botão
- Gerir a visibilidade -> por predefinição a visibilidade um espetáculo quando é definida como "não visível". Pode mudar esta informação escolhendo o espetáculo e clicando no botão

ATENÇÃO: Ao remover o espetáculo irá remover todas as reservas não pagas para esse espetáculo



FIM
