Licenciatura em Informática

- Análise E Conceção de Sistemas de Informação -

- Bases de Dados II -

- Tecnologias de Internet -

SiGREs - Sistema de Gestão de Reserva de Espaços

Documento de Projeto –Trabalho de Grupo

Realizado por:

Aluno Fábio Tordo, Nº150118013

Aluna Rossana Naki, Nº150118047

Aluno Rui Carvoeiro, Nº150118032

Santarém, 21 de janeiro de 2017

Índice

[Introdução 6](#_Toc472873766)

[Descrição do Sistema 7](#_Toc472873767)

[Principais Funções do Sistema 11](#_Toc472873768)

[Diagrama de Casos de Utilização 12](#_Toc472873769)

[Descrição dos Casos de Utilização 13](#_Toc472873770)

[Login 13](#_Toc472873771)

[Logout 14](#_Toc472873772)

[Listar Reservas 14](#_Toc472873773)

[Adicionar Reservas 15](#_Toc472873774)

[Excluir Reservas 16](#_Toc472873775)

[Confirmar a Reserva 17](#_Toc472873776)

[Gerir Sala 18](#_Toc472873777)

[Gerir Utilizadores 19](#_Toc472873778)

[Selecionar Diretamente Sala 20](#_Toc472873779)

[Pesquisar 21](#_Toc472873780)

[Característica 22](#_Toc472873781)

[Tempo de reserva 23](#_Toc472873782)

[E-mail 24](#_Toc472873783)

[Diagrama de Classes 25](#_Toc472873784)

[DER-Diagrama Entidade-Relacionamento 26](#_Toc472873785)

[Modelo/Esquema Lógico de Dados 27](#_Toc472873786)

[Modelo / Esquema Físico de Dados 28](#_Toc472873787)

[Interfaces do Utilizador 31](#_Toc472873788)

[I\_H: Homepage 31](#_Toc472873789)

[I\_PP: Ecrã Principal Professor (Listar Reservas) 31](#_Toc472873791)

[I\_PA: Ecrã Principal Administrador (Listar Reservas) 32](#_Toc472873793)

[I\_PR: Ecrã Principal Responsável Sala (Listar Reservas) 32](#_Toc472873795)

[I\_PF: Ecrã Principal Funcionário (Listar Reservas) 33](#_Toc472873797)

[I\_AR: Ecrã Adicionar Reservas (Opções Disponíveis) 33](#_Toc472873799)

[I\_ER: Ecrã Excluir Reservas 34](#_Toc472873801)

[I\_CR: Ecrã Confirmar Reservas 34](#_Toc472873803)

[I\_GR: Ecrã Gerir Salas 35](#_Toc472873805)

[I\_GU: Ecrã Gerir Utilizadores 35](#_Toc472873807)

[I\_GUA: Ecrã Gerir Utilizadores Administrador 36](#_Toc472873809)

[I\_RNP: Ecrã Reservas Não Pendentes 36](#_Toc472873811)

[I\_PS: Ecrã Pesquisar Sala 37](#_Toc472873813)

[Diagramas de Sequência 38](#_Toc472873815)

[Funcionalidade Login 38](#_Toc472873816)

[Funcionalidade Logout 38](#_Toc472873818)

[Funcionalidade Listar Reservas 39](#_Toc472873820)

[Funcionalidade Adicionar Reservas 40](#_Toc472873824)

[Funcionalidade Excluir Reservas 40](#_Toc472873826)

[Funcionalidade Confirmar a Reserva 41](#_Toc472873828)

[Funcionalidade Gerir Sala 41](#_Toc472873830)

[Funcionalidade Gerir Utilizadores 42](#_Toc472873832)

[Funcionalidade Selecionar Diretamente Sala 43](#_Toc472873836)

[Funcionalidade Pesquisar 43](#_Toc472873838)

[Funcionalidade Característica 44](#_Toc472873840)

[Funcionalidade Tempo de reserva 44](#_Toc472873842)

[Diagrama de Comunicação 45](#_Toc472873843)

[Login 45](#_Toc472873844)

[Logout 45](#_Toc472873846)

[Listar Reservas 45](#_Toc472873848)

[Adicionar Reservas 46](#_Toc472873850)

[Excluir Reservas 46](#_Toc472873852)

[Confirmar a Reserva 46](#_Toc472873854)

[Gerir Sala 46](#_Toc472873856)

[Gerir Utilizadores 47](#_Toc472873858)

[Diagrama de Estados 48](#_Toc472873860)

[Diagrama de Atividades 49](#_Toc472873861)

[Implementação dos Subprogramas Armazenados 50](#_Toc472873862)

[Funcionalidade 1 50](#_Toc472873863)

[Funcionalidade 2 51](#_Toc472873864)

[Funcionalidade 3 52](#_Toc472873865)

[Funcionalidade 4 52](#_Toc472873866)

[Funcionalidade 5 53](#_Toc472873867)

[Funcionalidade 6 54](#_Toc472873868)

[Funcionalidade 7 55](#_Toc472873869)

[Funcionalidade 8 55](#_Toc472873870)

[Funcionalidade 9 56](#_Toc472873871)

[Funcionalidade 10 57](#_Toc472873872)

[Funcionalidade 11 59](#_Toc472873873)

[Funcionalidade 12 60](#_Toc472873874)

[Funcionalidade 12 60](#_Toc472873875)

[Funcionalidade 13 e 14 61](#_Toc472873876)

[Funcionalidade 15 62](#_Toc472873877)

[Funcionalidade 16 62](#_Toc472873878)

[Funcionalidade 17 63](#_Toc472873879)

[Funcionalidade 18 64](#_Toc472873880)

[Funcionalidade 19 64](#_Toc472873881)

[Funcionalidade 20 65](#_Toc472873882)

[Funcionalidade 21 65](#_Toc472873883)

[Funcionalidade 22 66](#_Toc472873884)

[Funcionalidade 23 66](#_Toc472873885)

[Funcionalidade 24 67](#_Toc472873886)

[Funcionalidade 25 67](#_Toc472873887)

[Funcionalidade 26 68](#_Toc472873888)

[Funcionalidade 27 68](#_Toc472873889)

[Funcionalidade 28 69](#_Toc472873890)

[Funcionalidade 29, 30 e 31 70](#_Toc472873891)

[Funcionalidade 32 71](#_Toc472873892)

[Funcionalidade 33 72](#_Toc472873893)

[Funcionalidade 34 73](#_Toc472873894)

[Conclusões 74](#_Toc472873895)

[Bibliografia 75](#_Toc472873896)

Introdução

A elaboração deste trabalho tem como objetivo resolver um problema imposto pelos docentes, que consistia na falta que a faculdade sentia em ter um sistema funcional para gerir a ocupação das salas. Para isso é necessário a criação de um sistema de reservas para resolver essa situação. A solução do problema passa por três fases, primeiro faremos o planeamento do projeto, através de diagramas UML, seguidamente procedemos à criação da base de dados e por ultimo fizemos a implementação de um web site com todas as funcionalidades necessárias para o funcionamento do sistema. Para o desenvolvimento deste projeto guiamo-nos pelos conhecimentos adquiridos nas disciplinas de Análise e Conceção de Sistemas de Informação, Base de Dados 2 e Tecnologias da Internet.

Descrição do Sistema

A Escola Superior de Gestão e Tecnologia do Instituto Politécnico de Santarém necessita de um sistema de informação para controlar a disponibilidade de salas de aula e outros espaços. Até agora, o processo de reserva de sala, para as diversas atividades, tem sido feito manualmente: registada em agendas pessoais ou calendários de parede. Como podemos imaginar, há vários problemas decorrentes desses métodos. Essencialmente, torna-se muito complicado, para quem precisa de uma sala, saber se ela está disponível para uma atividade, numa data e hora específicas.

Inicialmente, tentando evitar esse problema, a direção da escola ordenou que o horário de ocupação da sala fosse afixado perto da porta da mesma. No entanto, outras questões foram despertadas: As atualizações regulares para os horários dos cursos, torna a informação rapidamente desatualizada. Então, quando alguém está à entrada da sala, a olhar para o calendário publicado lá, não sabe exatamente se a sala está livre ou não... Segundo o calendário publicado junto da sala ela poderia estar disponível naquele momento, mas alguém pode ter feito uma reserva dessa mesma sala, a iniciar dali por 5 minutos, junto de uma funcionária. Este tipo de informação é registado apenas numa agenda pessoal para os serviços administrativos da escola.

Se um professor quer saber se uma sala específica estará disponível daqui por 2 meses, é impossível sem falar com a pessoa responsável pela tarefa de gerir salas. Outro problema vem das horas de trabalho... Se alguém precisa dessas informações após as 18 horas, os serviços não estão disponíveis.

**a necessidade**

Assim, a necessidade deste sistema é apoiada em três pontos principais:

1. Integração - o sistema deve centralizar todas as informações sobre as reservas;
2. Disponibilidade - o sistema deve estar sempre disponível;
3. Segurança - o sistema deve controlar quem é concedida para fazer as diferentes operações e registrar quem fez as operações.

**1) Integração**

Os dados fundamentais deste novo sistema serão os horários dos cursos regulares, e em geral todos os cursos, na escola. Cursos como Gestão de Empresas, Contabilidade e Finanças, Marketing e Publicidade, Informática, Negócios Internacionais e Erasmus têm os seus próprios horários que ocupam as salas na maioria das vezes. Os cursos de Mestrado e os TeSP também ocupam regularmente as salas. As outras reservas são muito específicas e são feitas por professores para, eventualmente, ter uma reunião com os alunos, fazer um exame ou uma prova escrita, uma atividade de projeto, ou outro tipo de atividades que requeiram uma sala.

Os horários são produzidos pela Comissão de Horários e publicado antes do início de cada semestre. Após o início do semestre, pode haver atualizações para os horários dos cursos e haverá outra publicação, sempre que for necessário.

O sistema não permite a reserva no passado, ou seja, não podemos registar uma reserva com a data de ontem. Os Utilizadores podem inserir reservas para o ano/semestre letivo seguinte. O utilizador (professor ou funcionário) pode registrar a reserva de uma sala para um período específico de tempo. Haverá uma pessoa responsável para cada sala e, se ocorrer a reserva de sala por outro utilizador, entretanto, a pessoa responsável será sempre informada para poder analisar e aprovar. O sistema vai garantir a comunicação necessária entre os utilizadores interessados ​​nas reservas e as pessoas responsáveis ​​pelas salas.

O endereço de e-mail dos utilizadores deve ser registrado no sistema, pois toda a comunicação será enviada por aí. Na maioria das operações, o Sistema tem que enviar mensagens de e-mail pré-definidos para os usuários. O texto destas mensagens de e-mail é padrão e específico para o tipo de operação que o gerar. Algumas das mensagens pode incluir links para algumas das funcionalidades do sistema (confirmação da reserva, e assim por diante).

**2) Disponibilidade**

Dado que o sistema deve estar disponível todos os dias, todas as horas, em qualquer lugar, a Internet foi o meio ambiente selecionado.

**3) Segurança**

Por motivos de segurança, vários tipos de utilizadores serão considerados. Cada um dos quais tem um conjunto específico de funcionalidade para usar. O processo de autenticação será assegurado pelo próprio sistema. Esta funcionalidade é para ser considerada. As credenciais dos novos utilizadores devem ser carregadas no sistema por um administrador.

**OS UTILIZADORES**

Os principais utilizadores do sistema são, em termos funcionais de negócio: o Professor, o Funcionário do piso e o Responsável de sala. Em termos funcionais técnicos, temos a considerar um Administrador do sistema que gere as credenciais dos restantes.

**O Professor**

Em termos práticos, resumidamente, um Professor pode realizar 3 operações em torno das reservas de sala: Adicionar, Consultar e Remover. Pode ainda fazer pesquisas sobre as características e ocupação das salas.

**A) Adicionar nova reserva** – O registo de uma reserva pressupõe (A1) a escolha da sala e (A2) a definição do *timing* da reserva.

**A1)** Na escolha da sala, o Professor pode optar por selecionar diretamente a sala que pretende ou pode recorrer à pesquisa de sala disponível. Neste último caso, ao procurar por uma sala, o professor pode começar por indicar uma data e uma hora e, em seguida, o sistema mostra todas as salas livres (incluindo breve descrição das características). A descrição das aulas na lista será (SL1: 20 computadores, 30 lugares, e outro tipo de recursos - videoprojector, etc.).

Caso contrário, o professor pode escolher a data e hora e ainda pode indicar qual/ais o(s) tipo(s) de recursos deve(m) existir nessa sala (número de computadores, projetor de vídeo, número de lugares).

A lista de salas é apresentada com todas as salas disponíveis (e com as funcionalidades necessárias).

Em seguida, o professor seleciona apenas uma sala da lista e clica no botão de reserva.

Caso o professor já saiba a sala que pretenda, pode indicar diretamente qual é a sala e escolher para reservar.

**A2)** Na definição dos tempos das reservas, o professor pode optar por usar uma de três vias:

* **para um intervalo** - escolhe o dia, a sala, e define a hora de início e de fim da reserva;
* **por período** - escolhe o dia, a sala e o período: Manhã - das 8:30 até 13:30; Tarde - de 13:30 até 18:30, ou; Noite - a partir de 18:30 até 23:50. O sistema deve considerar automaticamente a hora de início e de fim);
* **e baseado num Padrão** - baseado num padrão repetitivo - uma sala, vários dias, um intervalo de horas e a regra de repetição do evento:

Independentemente da forma como o professor selecionou a sala podem ocorrer duas situações:

1. **(L)** se a sala estiver livre para o período definido, o sistema irá enviar automaticamente uma notificação por e-mail para o professor e para o responsável de sala, no momento em que o professor clica no botão da reserva. Esta mensagem "*A sua reserva foi gravada. Por favor, aguarde a mensagem de confirmação*" será enviada para o professor e uma outra mensagem, contendo um "Link Confirmação", será enviada a ambos. Em termos de sistema, a confirmação é feita pelo responsável quando clicar no link existente na mensagem (gerada aquando da reserva) que recebeu. Se isso acontecer, o processo fica concluído e o professor receberá a mensagem de confirmação: "*A sua reserva foi confirmada*". Se o responsável não concorda, outra mensagem será enviada (responsável pode propor uma sala diferente), com um texto diferente: "*A sua reserva não foi confirmada. Reservámos a sala* ***xpto*** *como uma alternativa*", ou "*Sua reserva não foi confirmada e nenhum espaço foi proposto. Entre em contato com os Serviços*". Neste último caso, uma sala nova será negociada fora do contexto do sistema e um dos utilizadores (professor ou responsável de sala) fará a reserva de novo.
2. **(O)** se a sala estiver ocupada, o sistema não permitirá a reserva de imediato e pode dar-se o caso de ter de ocorrer uma negociação (dentro e/ou fora do sistema). Por exemplo: o professor 1 regista no sistema uma reserva para a sala 1.3 para quarta-feira, 2 de novembro de 2016, 17:00-18:00. Posteriormente, o professor 2 entra no sistema e tenta reservar a mesma sala (1.3), no mesmo dia (quarta-feira, 2 de novembro de 2016) para as 17:30-18:30. Então, uma mensagem será mostrada ao professor 2: "*Esta sala, para esta data, está reservada pelo professor 1*" e há dois botões: "Continuar" e "Cancelar". Se o Professor 2 clicar em Cancelar, não há problema, pois assume-se que desistiu da reserva. Mas, se acionar o botão "Continuar", o sistema envia e-mail ao responsável ​​da sala informando da reserva em sobreposição. O responsável da sala terá de obter uma autorização do Professor 1 para realizar o cancelamento da reserva inicial. Se o professor 1 concordar, o responsável de sala exclui a reserva e confirma reserva do Professor 2. O Sistema envia e-mail com informações tanto para os professores (a reserva é excluída para o professor 1 e a reserva é confirmada para o professor 2). O responsável de sala, ou o próprio professor 1, pode posteriormente fazer uma nova reserva.

**B) Listar Reservas** – O Professor pode ver as suas próprias reservas e as respetivas descrições.

**C) Excluir Reservas** – O Professor pode excluir as suas próprias reservas a partir de lista de reserva. Sistema envia e-mail a informar o professor e o responsável da sala.

**O Responsável de sala**

O Responsável de sala pode realizar várias operações em torno das reservas de sala: Adicionar, Consultar, Remover e Confirmar reservas de outros (desde que sejam das salas que monitoriza). Pode ainda fazer pesquisas sobre as características e ocupação de todas as salas. Ao responsável de sala cabe ainda gerir as próprias salas, ou seja, criar novas salas (fica responsável automaticamente) ou inativar as salas (sob sua responsabilidade) que vierem a ficar inoperacionais.

**A) Adicionar nova reserva** – O processo é em tudo idêntico ao processo de reserva feito pelo professor, com a pequena diferença de que a reserva pode não ficar em nome do responsável de sala, mas sim no nome de outra pessoa (um docente, ou outra entidade – professor, funcionário, entidade externa, etc.). Se se aperceber de que a reserva apresenta conflitos com outras reservas, o responsável deverá “negociar”, fora do sistema, uma solução com as partes envolvidas. Só nesse momento é que registará no sistema a solução encontrada.

**B) Confirmar reservas** – Aquando de uma reserva, o Responsável tem a última palavra e deve confirmar a reserva através do link que consta do e-mail que recebeu a dar conta da reserva.

**C) Listar Reservas** – o Responsável pode ver uma lista de reserva para todas as salas. Responsável pode filtrar por professor e por sala.

**D) Excluir Reservas** – o Responsável pode excluir Reservas, sendo o professor (ou quem tenha realizado a reserva) notificado dessa ocorrência por e-mail.

**E) Gerir Salas** – o Responsável pode inserir novas salas para o sistema e configurar as instalações das salas novas ou já existentes.

**O Funcionário**

A) Lista Reservas – Os funcionários da escola podem ver lista de reservas. O filtro pode ser apenas por sala e data.

Principais Funções do Sistema

O sistema em análise tem…

… os seguintes atores:

* Professores (consegue listar reservas, adicionar reservas e excluir reservas)
* Funcionários (consegue listar reservas)
* Administradores(tem o poder total, isto é, listar reservas, adiciona reservas, confirma reservas, exclui reservas, gerir salas, gerir utilizadores)
* Responsável de Sala (consegue listar reservas, adiciona reservas, confirma reservas, exclui reservas e gere salas)

… as seguintes funções:

1ºAdicionar reservas;

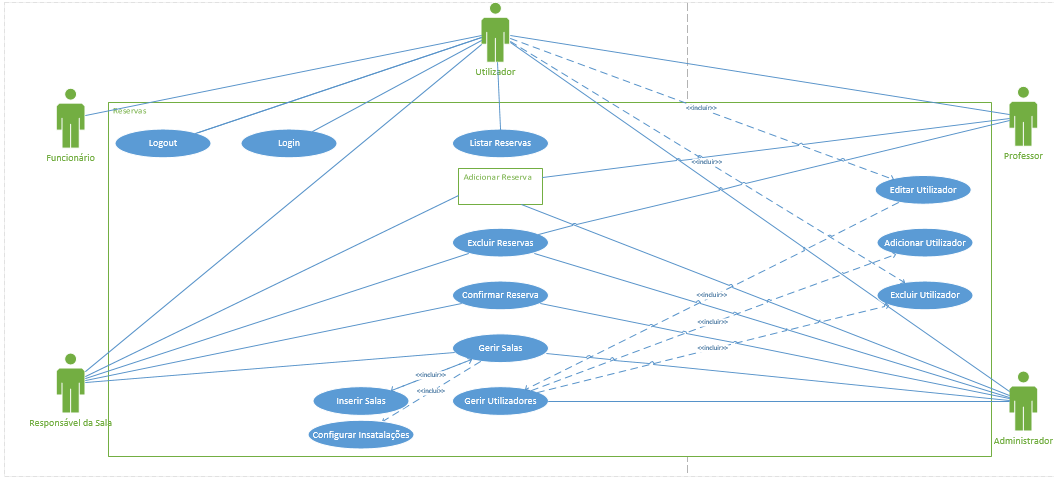
2ºListar reservas;

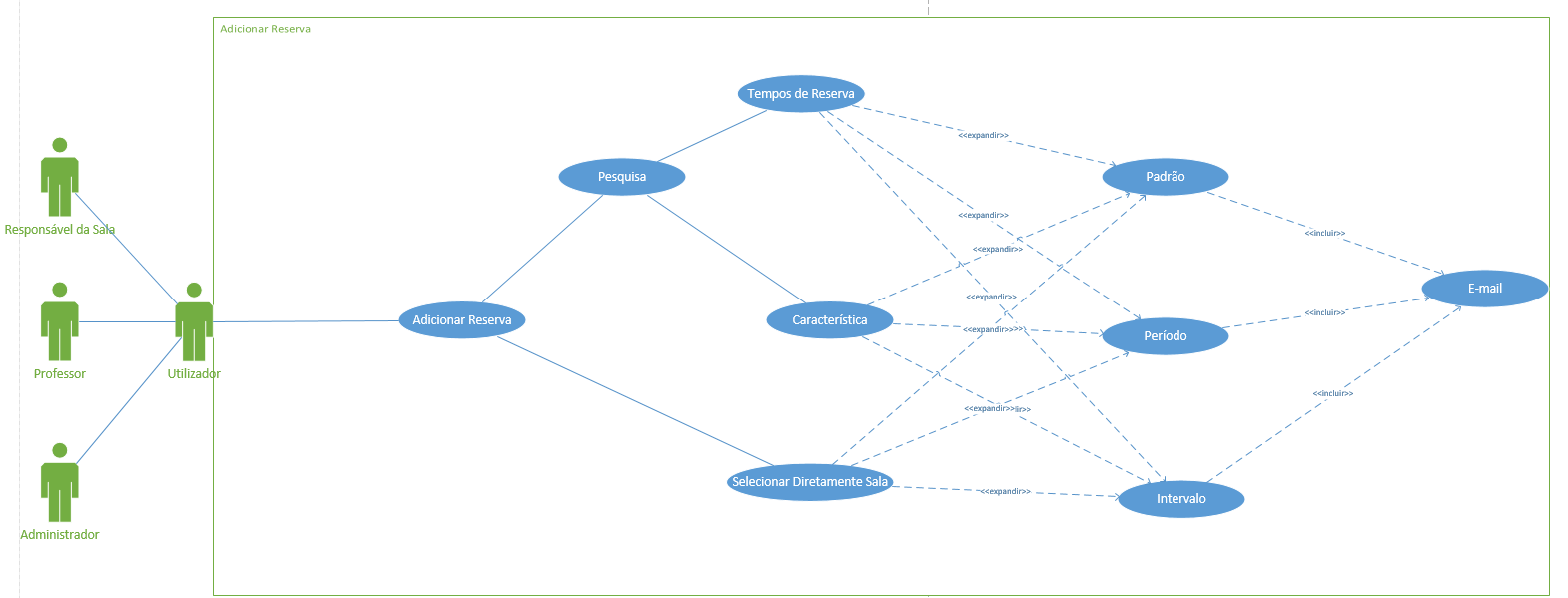
3ºExcluir reservas;

4ºConfirmar reservas;

5ºGerir salas;

Diagrama de Casos de Utilização





Descrição dos Casos de Utilização

|  |  |
| --- | --- |
| 1 - **Nome** | Login |

|  |  |
| --- | --- |
| 2 - **Atores** | Utilizador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Validar Utilizador |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H |

|  |
| --- |
| 5- **Cenário principal** |
| 1 – o utilizador (Funcionário, Responsável da Sala, Professor, Administrador), acede ao site;  2 -o sistema mostra um ecrã de validação [I\_H](Utilizador e Password);  3 -o utilizador insere as credencias e faz OK;  4 -o Sistema verifica se existem as credencias e apresenta o menu adequado ao tipo de utilizador (Funcionário, Responsável da Sala, Professor, Administrador); |

|  |
| --- |
| 6- **Cenário Alternativo – Erro password** |
| 4 -caso as credencias não existam, o utilizador é informado com uma mensagem e é dada nova possibilidade de entrar no sistema |

|  |  |
| --- | --- |
| 7- **Observações** | Á terceira tentativa de entrada sem que a password esteja correta, o utilizador será bloqueado e permanecerá assim durante 5 minutos. |

|  |  |
| --- | --- |
| 8- **Pós condições** | O utilizador é identificado |

|  |  |
| --- | --- |
| 1 - **Nome** | Logout |

|  |  |
| --- | --- |
| 2 - **Atores** | Utilizador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Desativar o utilizador |

|  |  |
| --- | --- |
| 4- **Ecrãs** | Todos os escrãs |

|  |  |
| --- | --- |
| 5- **Pré condições** | O utilizador tem a sessão ativa (fez o login corretamente) |

|  |
| --- |
| 6- **Cenário principal** |
| 1 – o utilizador (Funcionário, Responsável da Sala, Professor, Administrador) aciona a função de Logout;  2 – o sistema remove toda a informação da sessão;  3 – o sistema mostra o ecrã para efetuar um novo login. |

|  |  |
| --- | --- |
| 7- **Pós condições** | A sessão do utilizador é terminada |

|  |  |
| --- | --- |
| 1 - **Nome** | Listar Reservas |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Funcionário ou Responsável de Sala ou Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Consultar todas as reservas |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PP, I\_PA, I\_PR, I\_PF |

|  |  |
| --- | --- |
| 5- **Pré condições** | - O utilizador (Professor ou Funcionário ou Responsável de Sala ou Administrador) faz o login;  - O sistema disponibiliza todas as Reservas efetuadas num determinado dia/hora/semana, com predefinição no dia atual. |

|  |
| --- |
| 6- **Cenário principal** |
| 1 – O utilizador (Funcionário, Responsável da Sala, Professor, Administrador) faz login na página inicial;  2 – O utilizador visualiza as suas reservas; |

|  |  |
| --- | --- |
| 7- **Pós condições** | As reservas existentes são apresentadas no ecrã. |

|  |  |
| --- | --- |
| 1 - **Nome** | Adicionar Reservas |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Responsável de sala ou Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Criar uma ou mais reservas; |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PP, I\_PA, I\_PR, I\_AR |

|  |  |
| --- | --- |
| 5- **Pré condições** | - O utilizador (Professor ou Responsável de Sala ou Administrador) faz o Login.  - O Responsável de Sala só pode adicionar reservas das salas que é responsável. |

|  |
| --- |
| 6- **Cenário principal** |
| 1 – O utilizador (Professor ou Responsável de Sala ou Administrador) faz login na página inicial;  2 – O utilizador aciona a opção Adicionar Reservas na barra lateral da página;  3- São apresentadas duas opções:   * Selecionar diretamente as salas; * Pesquisa;   4 - O utilizador escolhe a opção que desejar e preenche os formulários apresentados;  5 - São apresentados no ecrã os resultados da pesquisa efetuada;  6 – O utilizador seleciona uma ou mais salas resultantes da pesquisa e clica no botão “reservar”. |

|  |  |
| --- | --- |
| 7- **Pós condições** | As reservas existentes são apresentadas no ecrã. |

|  |  |
| --- | --- |
| 1 - **Nome** | Excluir Reservas |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Responsável de sala ou Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Eliminar uma ou mais reservas do sistema. |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PP, I\_PA, I\_PR, I\_ER |

|  |  |
| --- | --- |
| 5- **Pré condições** | - O Professor só pode excluir as suas próprias reservas.  - Tem de existir reservas já efetuadas no sistema.  - O Responsável de Sala só pode excluir reservas das salas que tem poder. |

|  |
| --- |
| 6- **Cenário principal** |
| 1 - O utilizador faz login (Professor ou Responsável de Sala ou Administrador na página inicial;  2 – O utilizador aciona a opção Excluir Reservas na barra lateral da página;  3- O sistema apresenta uma lista com a/as reserva/as do utilizador (Professor ou Responsável), exceto o Administrador que pode listar todas as que pretende;  4- O utilizador seleciona a/as reserva/as a eliminar;  5- O utilizador clica no botão “excluir”;  6- O sistema apresenta uma mensagem a dizer que a ação foi concluída. |

|  |  |
| --- | --- |
| 7- **Pós condições** | As reservas selecionadas são excluídas com êxito. |

|  |  |
| --- | --- |
| 1 - **Nome** | Confirmar a Reserva |

|  |  |
| --- | --- |
| 2 - **Atores** | Administrador ou Responsável da Sala |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Confirmar as reservas pendentes |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PA, I\_PR, I\_CR |

|  |  |
| --- | --- |
| 5- **Pré condições** | -O Responsável da Sala tem de ter os direitos necessários sobre a sala em especifico para confirmar a sua reserva. |

|  |
| --- |
| 6- **Cenário principal** |
| 1 - O utilizador (Responsável de Sala ou Administrador) faz login;  2 – O utilizador aciona a opção Confirmar Reservas na barra lateral da página;  3 - o sistema recolhe todas as reservas pendentes;  4- o utilizador (Administrador, Responsável da Sala) escolhe as reservas pendentes que pretende confirmar e faz OK;  5- o sistema guarda as reservas confirmadas |

|  |  |
| --- | --- |
| 7 - **Pós condições** | As salas que estavam pendentes no sistema, passam a ser confirmadas |

|  |  |
| --- | --- |
| 1 - **Nome** | Gerir Sala |

|  |  |
| --- | --- |
| 2 - **Atores** | Administrador ou Responsável de Sala |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Adicionar uma nova Sala e configurar as suas instalações |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PA, I\_PR, I\_GS |

|  |  |
| --- | --- |
| 5- **Pré condições** | -O Responsável da Sala tem de ter os direitos necessários sobre a sala em especifico para confirmar a sua reserva. |

|  |
| --- |
| 6- **Cenário principal** |
| 1 - O utilizador (Responsável de Sala ou Administrador) faz login;  2 – O utilizador aciona a opção Gerir Sala na barra lateral da página;  3 - São apresentadas três opções:   * Inserir uma nova sala; * Excluir sala; * Configurar uma sala já existente;   4 - O utilizador escolhe a opção que desejar e preenche os formulários apresentados;  5 – O utilizador carrega “Ok” após preenchimento dos campos necessários; |

|  |  |
| --- | --- |
| 7- **Observações** | Para se configurar uma sala é necessário indicar qual a sala a alterar |

|  |  |
| --- | --- |
| 8- **Pós condições** | As características da(s) sala(s) foram alteradas. |

|  |  |
| --- | --- |
| 1 - **Nome** | Gerir Utilizadores |

|  |  |
| --- | --- |
| 2 - **Atores** | Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | -Editar Utilizador, adicionar novo Utilizador, excluir Utilizador |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PA, I\_GU |

|  |
| --- |
| 5- **Cenário principal** |
| 1 – O Admistrador faz login na página inicial;  2 – O Admistrador aciona a opção Gerir Utilizadores na barra lateral da página;  3- São apresentadas três opções:   * Editar Utilizador; * Adicionar Novo Utilizador; * Excluir Utilizador.   4 - O Admistrador escolhe a opção que desejar e preenche os formulários apresentados e pressiona “Ok”; |

|  |  |
| --- | --- |
| 7- **Observações** | A opção Editar Utilizador é visível para todos os Utilizadores (Funcionário, Responsável da Sala, Professor, Administrador), isto é, todos os Utilizadores podem editar-se a si próprios exceto os Administradores que podem editar todos incluindo eles mesmos. |

|  |  |
| --- | --- |
| 8- **Pós condições** | O Utilizador (Funcionário, Responsável da Sala, Professor, Administrador) é alterado. |

|  |  |
| --- | --- |
| 1 - **Nome** | Selecionar Diretamente Sala |

|  |  |
| --- | --- |
| 2 - **Atores** | Responsável de Sala ou Admistrador ou Professor |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Escolher Diretamente a Sala |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PP, I\_PA, I\_PR, I\_RNP |

|  |
| --- |
| 5- **Cenário principal** |
| 1 – O utilizador (Professor ou Responsável de Sala ou Administrador) faz login na página inicial;  2 – O utilizador aciona a opção Adicionar Reservas na barra lateral da página;  3- São apresentadas duas opções:   * Selecionar diretamente as salas; * Pesquisar;   4 - O utilizador escolhe o método de pesquisa: selecionar diretamente as salas;  5 - São apresentados no ecrã os resultados das salas disponíveis para reserva;  6 – O utilizador seleciona uma ou mais salas resultantes da pesquisa e clica no botão “reservar”. |

|  |  |
| --- | --- |
| 8- **Pós condições** | A reserva irá ficar pendente |

|  |  |
| --- | --- |
| 1 - **Nome** | Pesquisar |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Responsável de sala ou Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Pesquisar as salas que estiverem disponíveis. |

|  |  |
| --- | --- |
| 4- **Ecrãs** | I\_H, I\_PP, I\_PA, I\_PR, I\_AR, I\_PS |

|  |  |
| --- | --- |
| 5- **Pré-condições** | - O utilizador fez o login ou com uma conta de professor, de responsável de sala ou de administrador  - Tem de existir reservas já efetuadas no sistema. |

|  |
| --- |
| 6- **Cenário principal** |
| 1 – O utilizador faz login na página inicial;  2 – O utilizador aciona a opção Adicionar Reservas na barra lateral da página;  3 -São apresentadas duas opções:   * Selecionar diretamente Sala; * Pesquisar.   5- O utilizador escolhe pesquisar.  6- Após a pesquisa estar completa, o utilizador escolhe uma ou mais salas das que foram apresentadas;  7-O utilizador clica no botão reservar; |

|  |  |
| --- | --- |
| 7- **Pós condições** | São apresentadas no ecrã todas as salas disponíveis dependendo do tipo de pesquisa escolhido. |

|  |  |
| --- | --- |
| 1 - **Nome** | Característica |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Responsável de sala ou Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Pesquisar as salas que estiverem disponíveis com as características pretendidas. |

|  |  |
| --- | --- |
| 4- **Pré condições** | - O utilizador fez o login ou com uma conta de professor, de responsável de sala ou de administrador;  - Tem de existir reservas já efetuadas no sistema. |

|  |
| --- |
| 5- **Cenário principal** |
| 1 – O utilizador faz login na página inicial;  2 – O utilizador aciona a opção Adicionar Reservas na barra lateral da página;  3 -São apresentadas duas opções:   * Selecionar diretamente Sala; * Pesquisar.   4- O utilizador escolhe pesquisar e são apresentadas outras duas opções:   * Pesquisa por características; * Pesquisa por tempos de reservas.   5- O utilizador escolhe o método de pesquisa: por características;  6- É apresentado um formulário para o utilizador definir a data, a hora e as características desejadas;  7- Após a pesquisa estar completa, o utilizador escolhe uma ou mais salas das que foram apresentadas;  8- O utilizador clica no botão reservar; |

|  |  |
| --- | --- |
| 6- **Pós condições** | São apresentadas no ecrã todas as salas disponíveis com o tipo de características escolhidas; |

|  |  |
| --- | --- |
| 1 - **Nome** | Tempo de reserva |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Responsável de sala ou Administrador |

|  |  |
| --- | --- |
| 3 – **Objetivos** | Pesquisar as salas que estiverem disponíveis. |

|  |  |
| --- | --- |
| 4- **Pré condições** | - O utilizador fez o login ou com uma conta de professor, de responsável de sala ou de administrador;  - Tem de existir reservas já efetuadas no sistema. |

|  |
| --- |
| 5- **Cenário principal** |
| 1 – O utilizador faz login na página inicial;  2 – O utilizador aciona a opção Adicionar Reservas na barra lateral da página;  3 -São apresentadas duas opções:   * Selecionar diretamente Sala; * Pesquisar.   4- O utilizador escolhe pesquisar e são apresentadas outras duas opções:   * Pesquisa por características; * Pesquisa por tempos de reservas.   5- O utilizador escolhe o método de pesquisa: por tempos de reserva;  6- São apresentadas outras três opções:   * Por intervalo; * Por período; * Por padrão.   7- O utilizador escolhe o método de pesquisa: por tempo de reserva;  7- É apresentado um formulário para o utilizador definir a data, a hora;  9- Após a pesquisa estar completa, o utilizador escolhe uma ou mais salas das que foram apresentadas;  10- O utilizador clica no botão reservar; |

|  |  |
| --- | --- |
| 6- **Pós condições** | São apresentadas no ecrã todas as salas disponíveis dependendo do tipo de pesquisa escolhido. |

|  |  |
| --- | --- |
| 1 - **Nome** | E-mail |

|  |  |
| --- | --- |
| 2 - **Atores** | Professor ou Responsável de Sala ou Administrador |

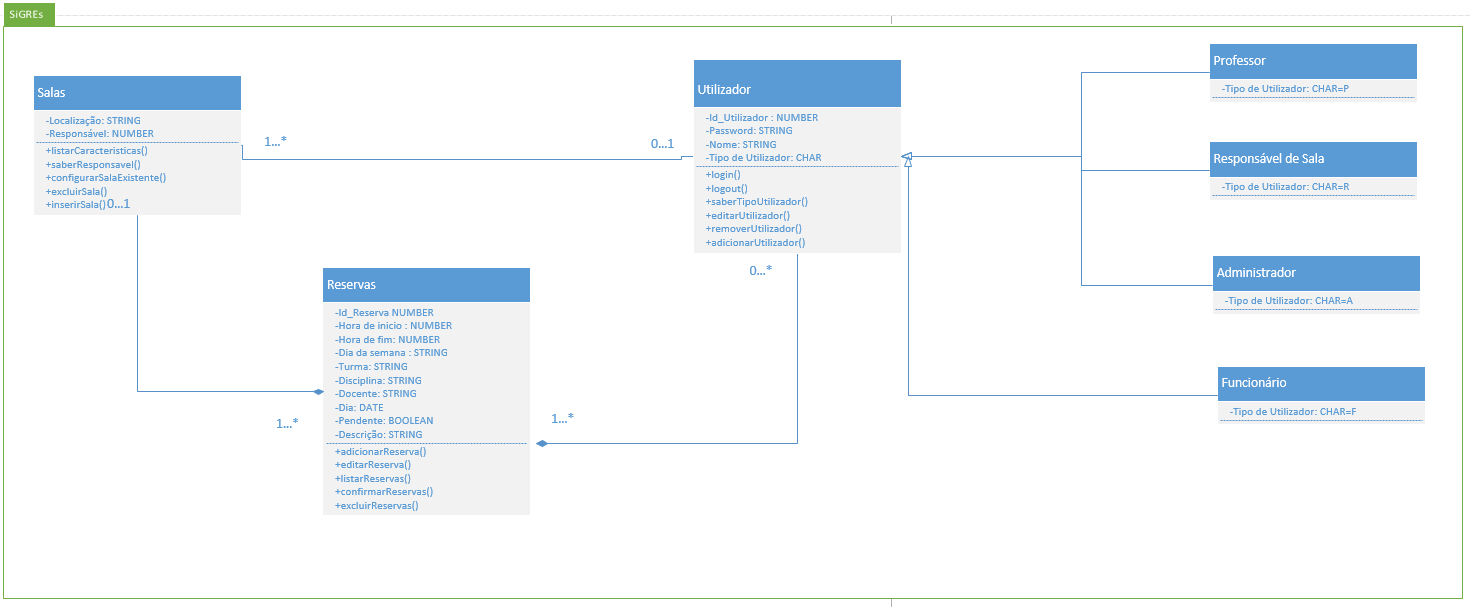
|  |  |
| --- | --- |
| 3 – **Objetivos** | -O Utilizador (Professor ou Responsável de Sala ou Administrador) que faz a reserva, indica ao sistema para enviar email ao Responsável de Sala para indicar a(s) reserva(s) efetuadas.  -O Responsável de Sala, após definir se aceita a reserva ou não, obriga que o sistema envie um outro email para o Utilizador que fez a reserva a indicar a sua resposta. |

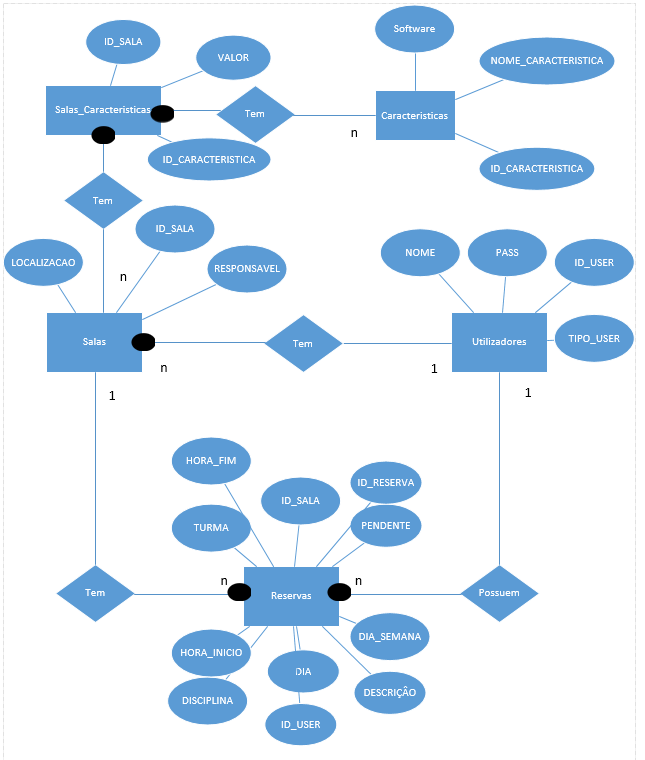
|  |  |
| --- | --- |
| 4- **Pré condições** | - É necessário Adicionar Reservas |

|  |
| --- |
| 5- **Cenário principal** |
| 1 – O utilizador faz login na página inicial;  2 – O utilizador aciona a opção Adicionar Reservas na barra lateral da página;  3 – Após Adicionar a Reserva o sistema envia um email para o Responsável de Sala em questão se confirma ou não a reserva  4- Depois do Responsável de Sala aceitar/recusar o pedido, é enviado um outro email para o Utilizador que reservou a sala. |

|  |  |
| --- | --- |
| - **Pós condições** | - A reserva da sala fica pendente ou fica |

Diagrama de Classes



DER-Diagrama Entidade-Relacionamento

Modelo/Esquema Lógico de Dados

**Tabela Salas**

(#ID\_SALA, LOCALIZACAO, RESPONSÀVEL)

**Tabela Características**

(#ID\_CARACTERISTICA, NOME\_CARACTERISTICA, SOFTWARE)

**Tabela Salas\_Caracteristicas**

(#ID\_SALA, #ID\_CARACTERISTICA, VALOR)

**Tabela Utilizadores**

(#ID\_USER, PALAVRA\_CHAVE, NOME, TIPO\_USER)

**Tabela Reservas**

(#ID\_RESERVA, ID\_SALA, HORA\_INICIO, HORA\_FIM, DIA\_SEMANA, TURMA, DISCIPLINA, DIA, PENDENTE, DOCENTE, DESCRIÇAO)

Modelo / Esquema Físico de Dados

CREATE TABLE Salas(

ID\_SALA NUMBER(4) PRIMARY KEY,

LOCALIZACAO NVARCHAR2(6)NOT NULL,

RESPONSAVEL NUMBER(4) NOT NULL

);

CREATE TABLE Carateristicas(

ID\_CARATERISTICA NUMBER(3) PRIMARY KEY,

NOME\_CARATERISTICA NVARCHAR2(20),

SOFTWARE CHAR(1) CHECK (SOFTWARE IN (0, 1))

);

CREATE TABLE Salas\_Carateristicas(

SALA NUMBER(4),

CARATERISTICA NUMBER(3),

VALOR NUMBER(2)

);

CREATE TABLE Utilizadores(

ID\_UTILIZADOR NUMBER(4) PRIMARY KEY,

NOME NVARCHAR2(60) NOT NULL,

PALAVRA\_PASSE NVARCHAR2(20) NOT NULL CHECK(LENGTH(PALAVRA\_PASSE) > 2),

EMAIL VARCHAR2(30) NOT NULL UNIQUE,

TIPO\_USER CHAR(1) CHECK (TIPO\_USER = 'P' OR TIPO\_USER = 'R' OR TIPO\_USER = 'F' OR TIPO\_USER = 'A')

);

CREATE TABLE Reservas(

ID\_RESERVA NUMBER(6) PRIMARY KEY,

SALA NUMBER(4) NOT NULL,

HORA\_INICIO NUMBER(4) NOT NULL,

HORA\_FIM NUMBER(4) NOT NULL,

DIA\_SEMANA NVARCHAR2(3),

TURMA NVARCHAR2(30),

DISCIPLINA NVARCHAR2(30),

DOCENTE NVARCHAR2(40),

DIA DATE,

PENDENTE CHAR(1) CHECK (PENDENTE IN (0, 1)), --NAO ACEITA BOOLEAN NEM BIT?

DESCRICAO NVARCHAR2(50)

);

ALTER TABLE Reservas ADD CONSTRAINT RESERVA\_SALA FOREIGN KEY (SALA) REFERENCES Salas(ID\_SALA);

ALTER TABLE Salas ADD CONSTRAINT SALA\_RESPONSAVEL FOREIGN KEY (RESPONSAVEL) REFERENCES Utilizadores(ID\_UTILIZADOR);

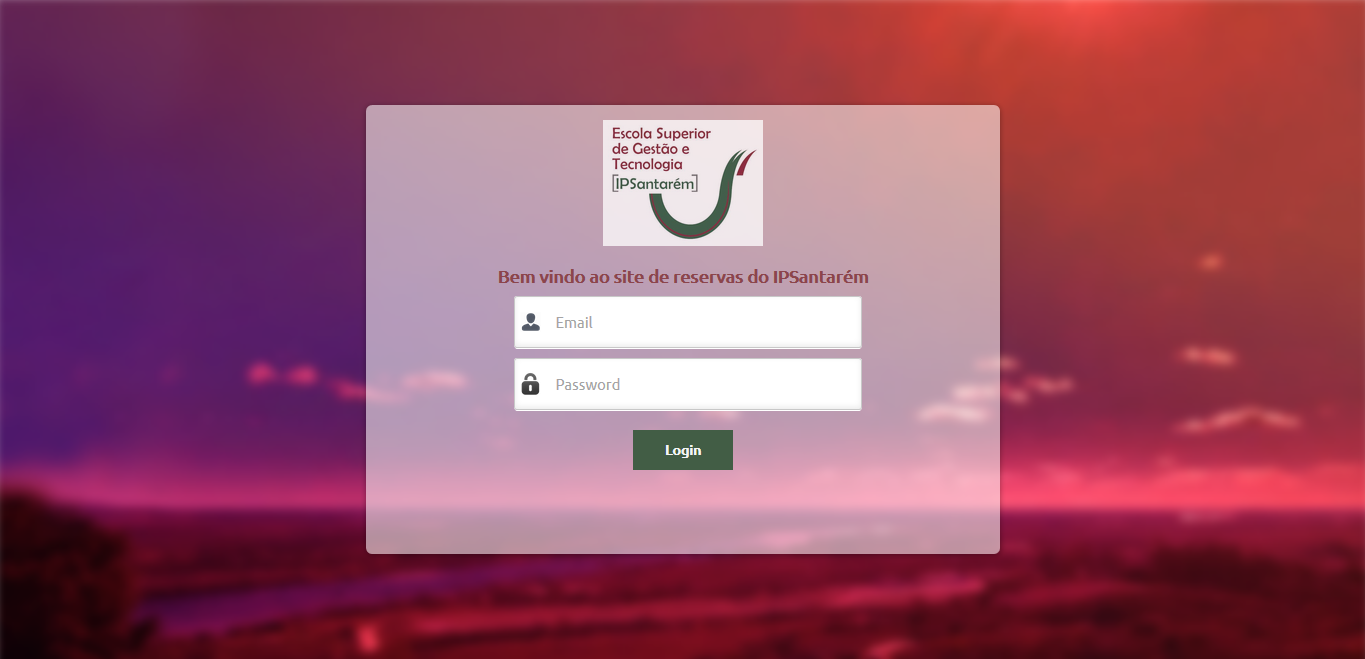
ALTER TABLE Salas\_Carateristicas ADD CONSTRAINT PK\_Salas\_Carateristicas PRIMARY KEY (CARATERISTICA, SALA);

ALTER TABLE Salas\_Carateristicas ADD CONSTRAINT Qual\_CARATERISTICA FOREIGN KEY (CARATERISTICA) REFERENCES Carateristicas(ID\_CARATERISTICA);

ALTER TABLE Salas\_Carateristicas ADD CONSTRAINT Qual\_SALA FOREIGN KEY (SALA) REFERENCES Salas(ID\_SALA);

Interfaces do Utilizador

I\_H: Homepage



I\_PP: Ecrã Principal Professor (Listar Reservas)



I\_PA: Ecrã Principal Administrador (Listar Reservas)



I\_PR: Ecrã Principal Responsável Sala (Listar Reservas)



I\_PF: Ecrã Principal Funcionário (Listar Reservas)



I\_AR: Ecrã Adicionar Reservas (Opções Disponíveis)



I\_ER: Ecrã Excluir Reservas



I\_CR: Ecrã Confirmar Reservas



I\_GR: Ecrã Gerir Salas



I\_GU: Ecrã Gerir Utilizadores



I\_GUA: Ecrã Gerir Utilizadores Administrador



I\_RNP: Ecrã Reservas Não Pendentes

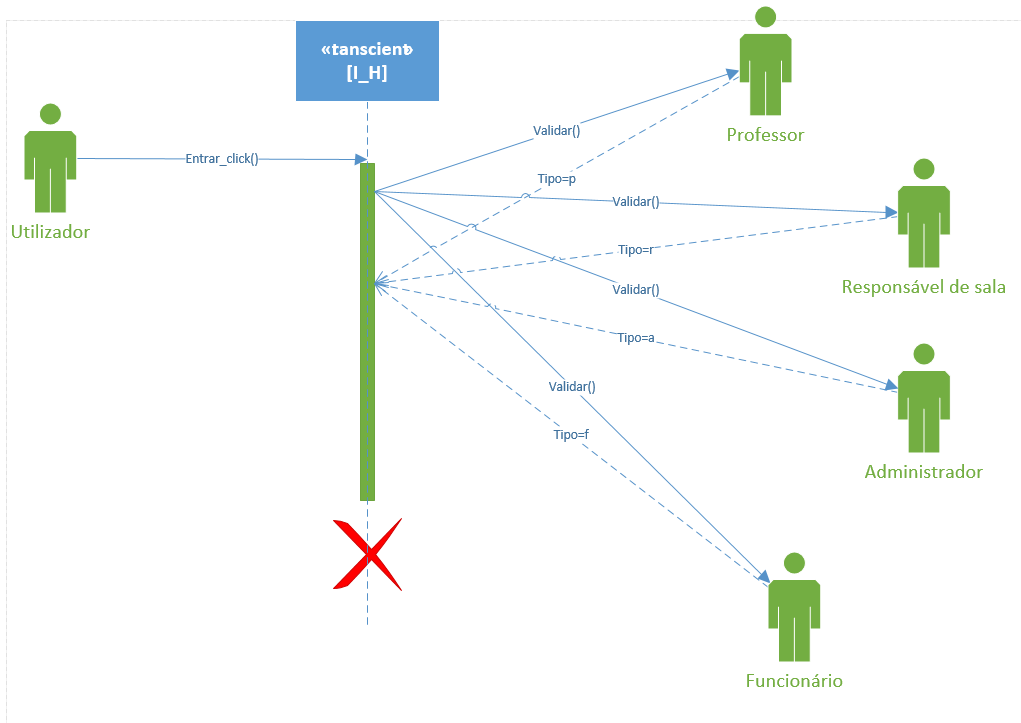


I\_PS: Ecrã Pesquisar Sala

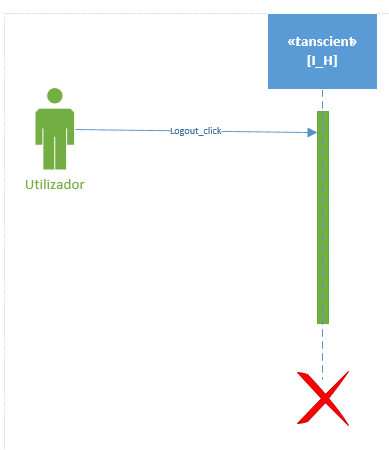


Diagramas de Sequência

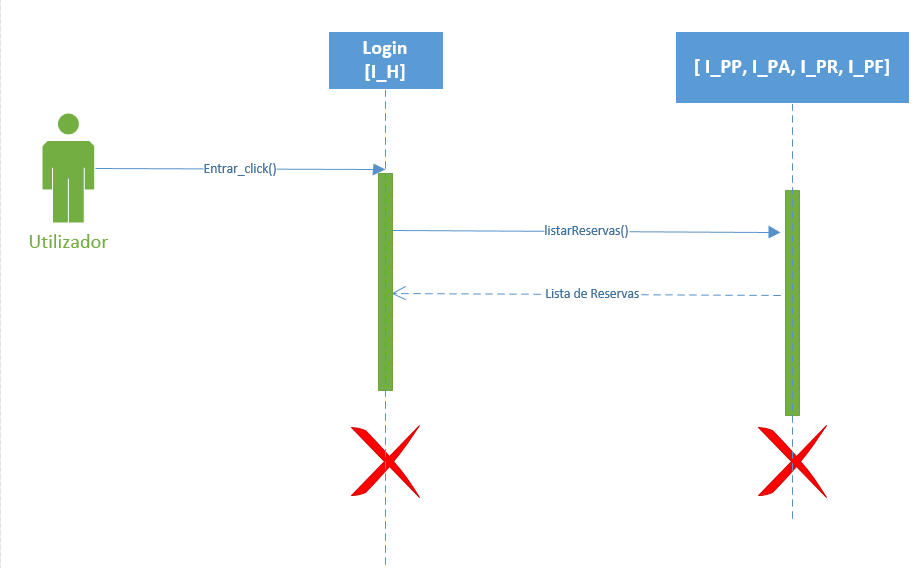
Funcionalidade Login



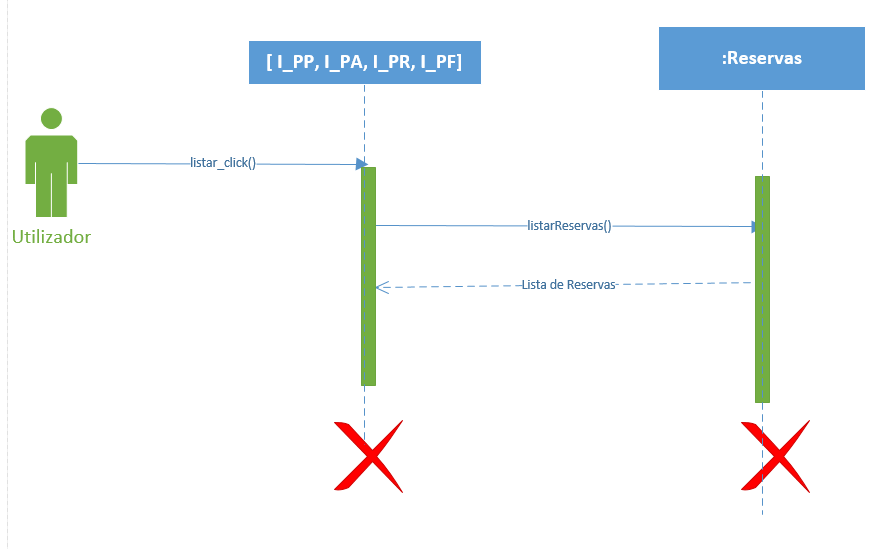
Funcionalidade Logout



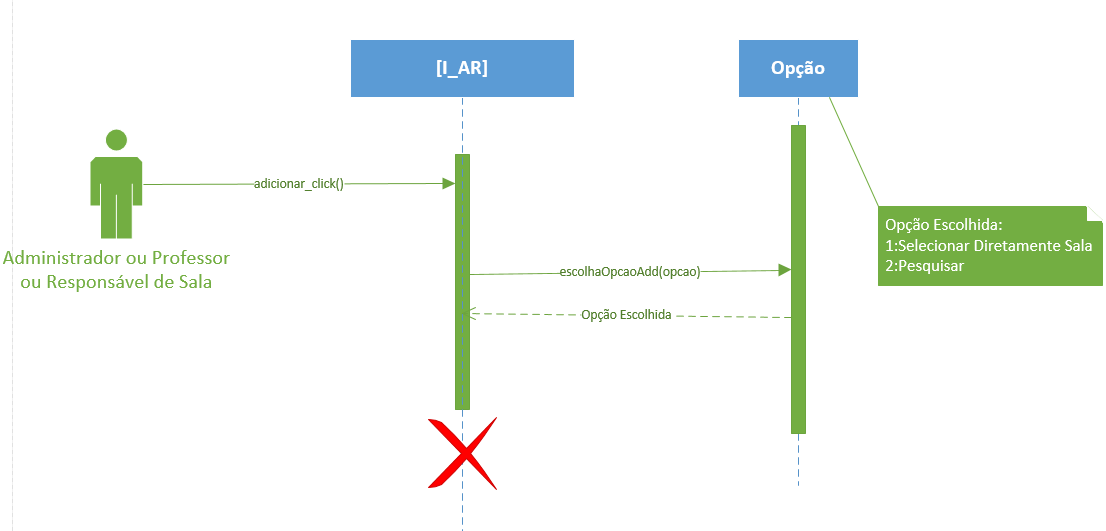
Funcionalidade Listar Reservas



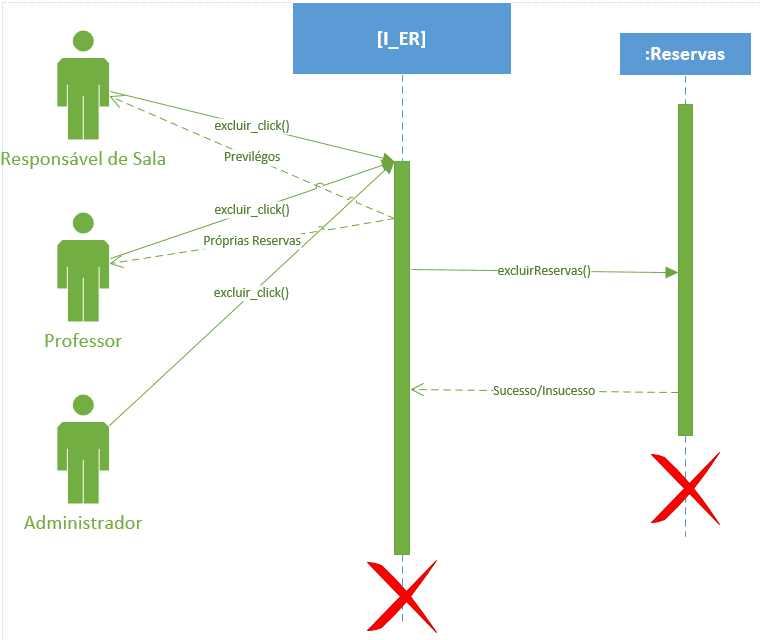
OU



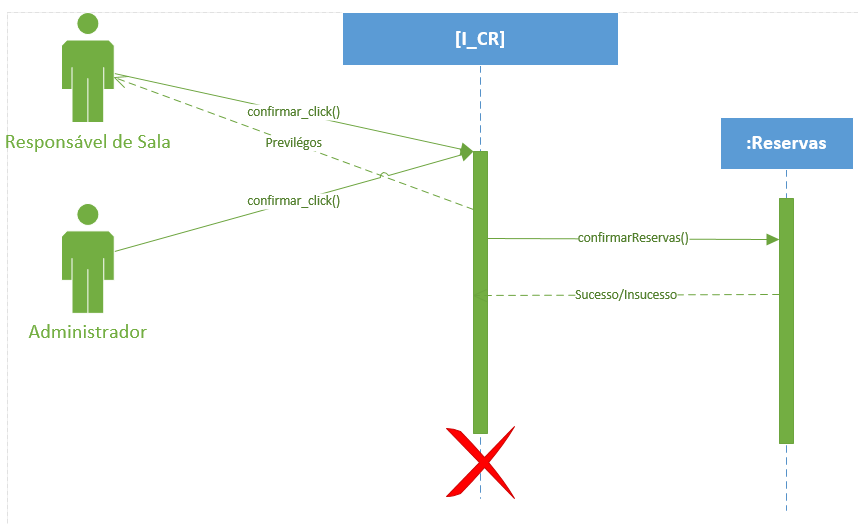
Funcionalidade Adicionar Reservas



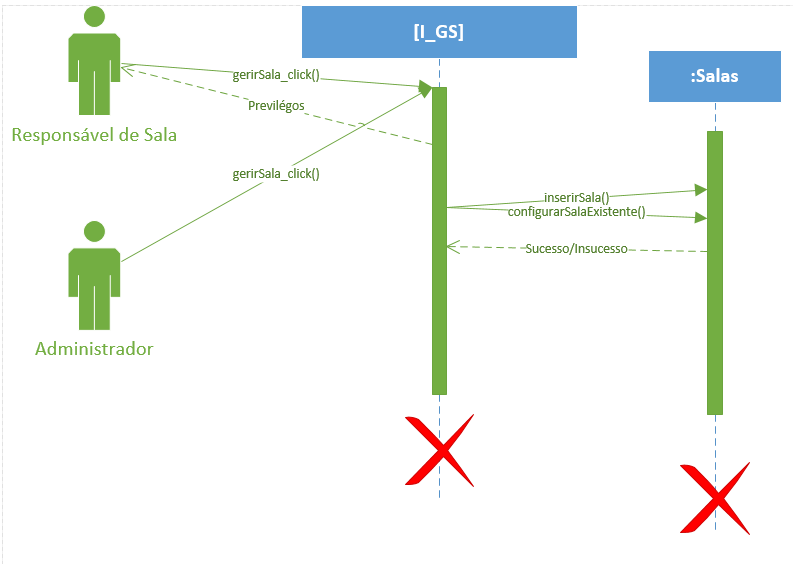
Funcionalidade Excluir Reservas



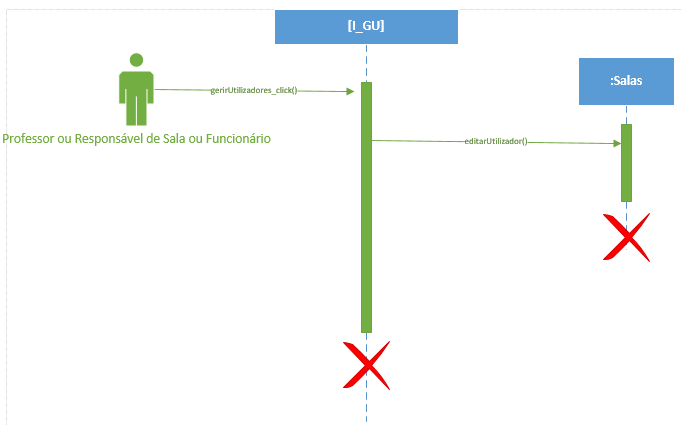
Funcionalidade Confirmar a Reserva



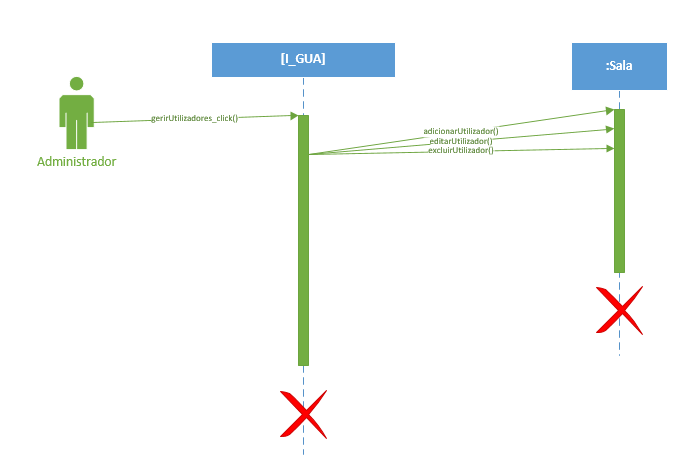
Funcionalidade Gerir Sala



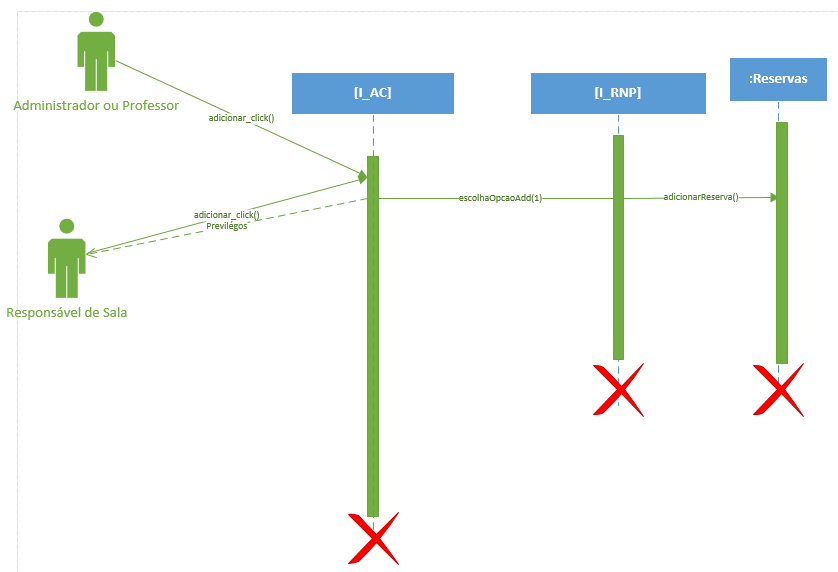
Funcionalidade Gerir Utilizadores



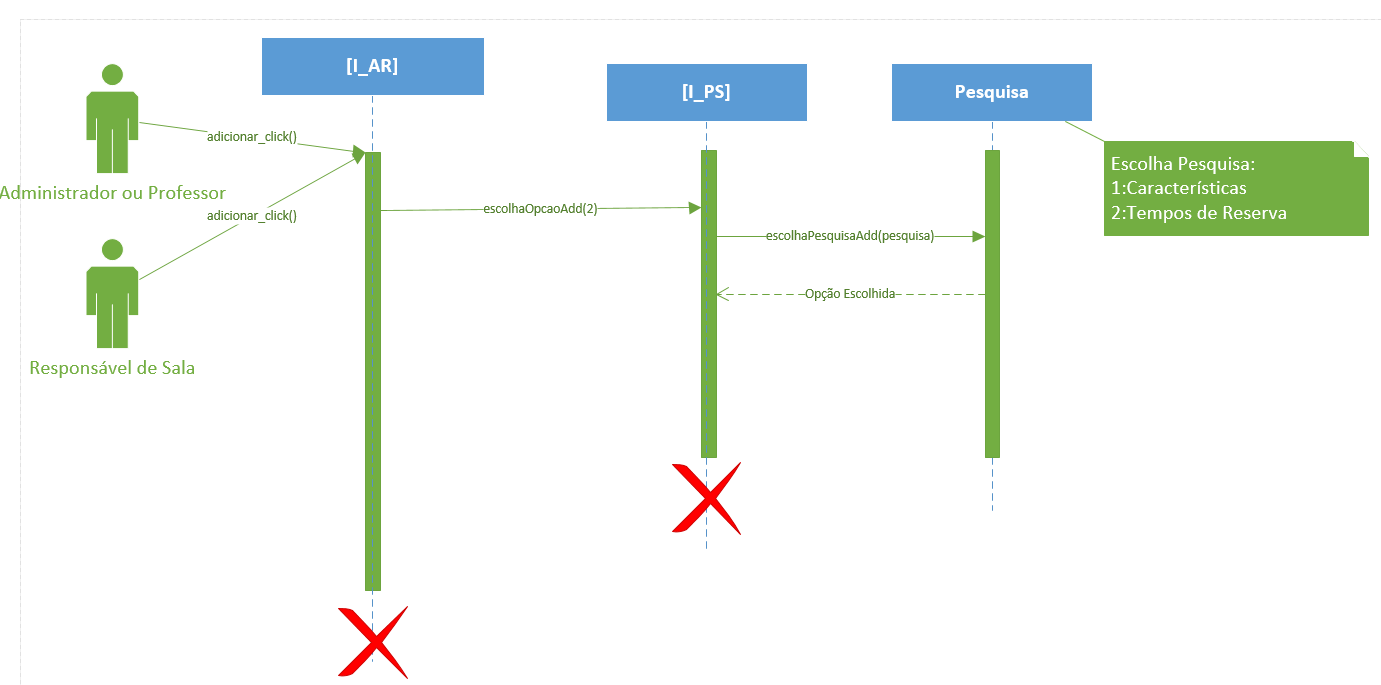
OU



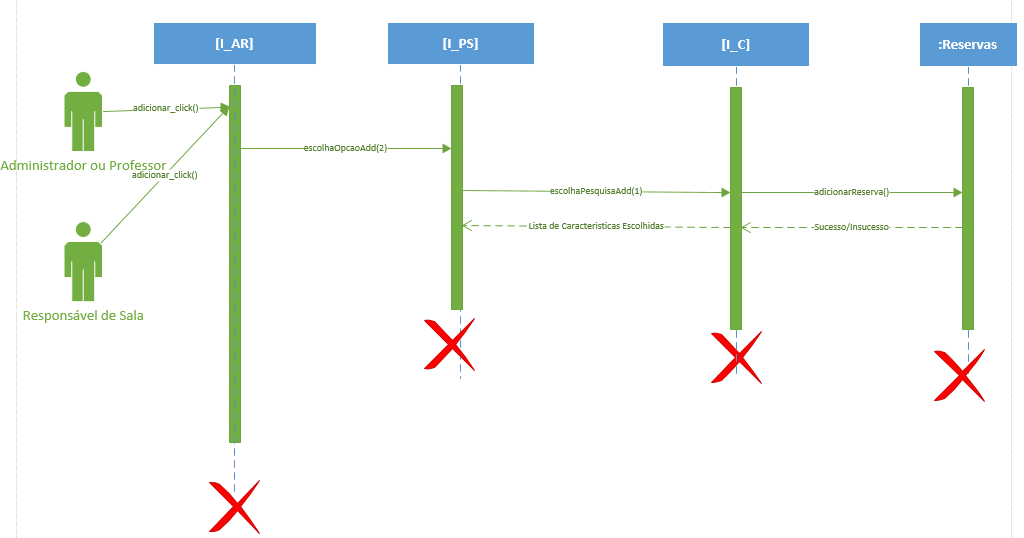
Funcionalidade Selecionar Diretamente Sala



Funcionalidade Pesquisar



Funcionalidade Característica



Funcionalidade Tempo de reserva

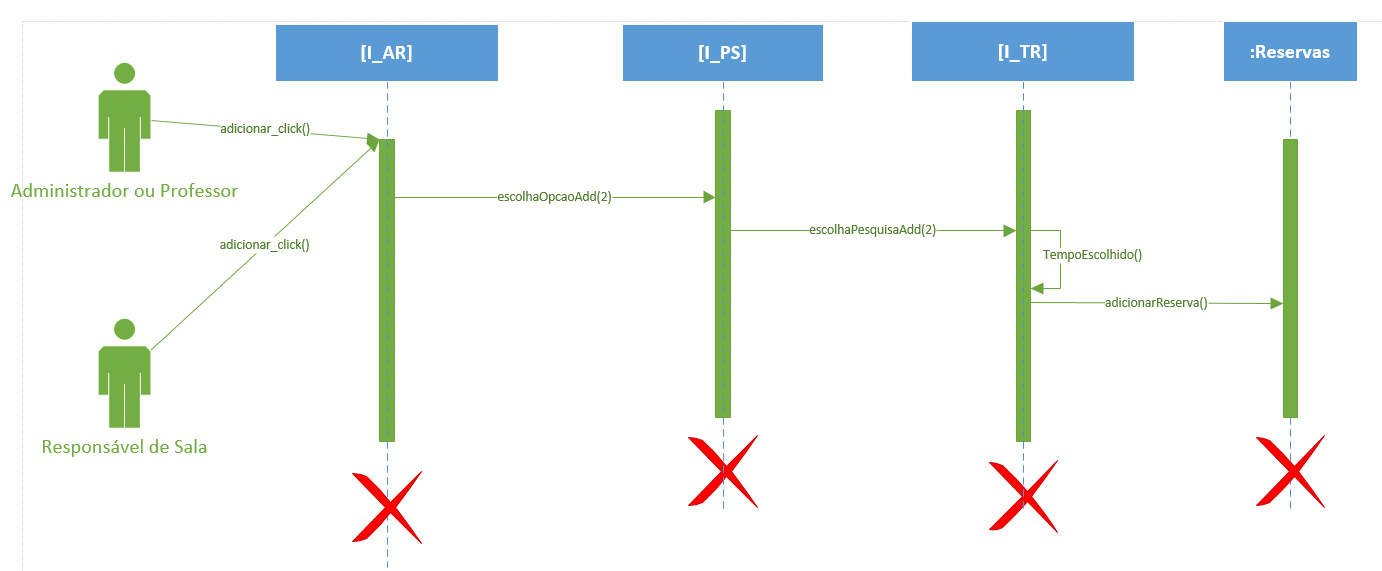
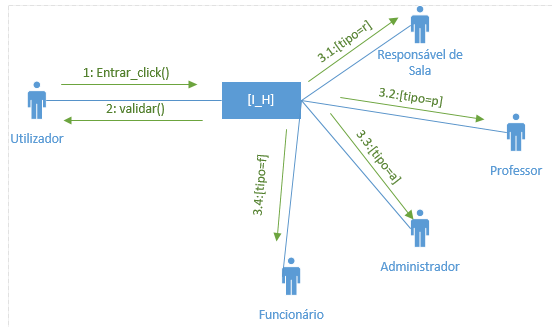
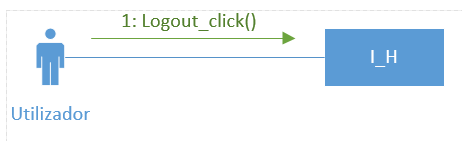


Diagrama de Comunicação

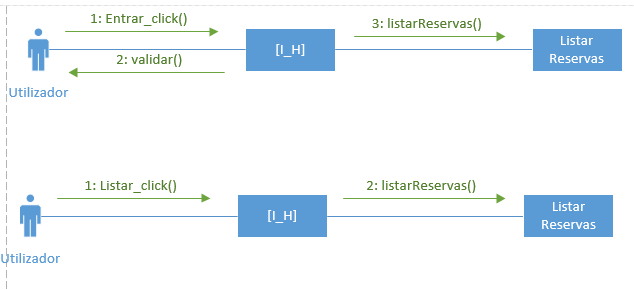
Login



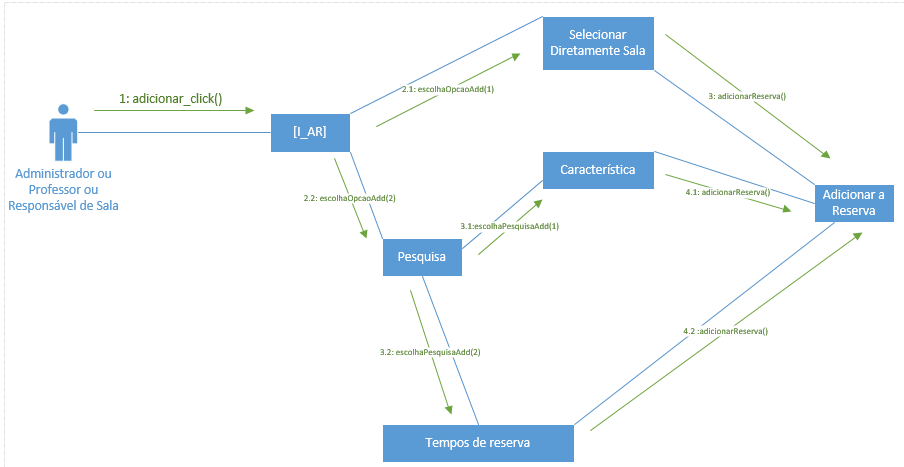
Logout



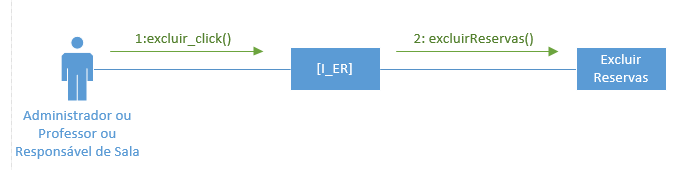
Listar Reservas



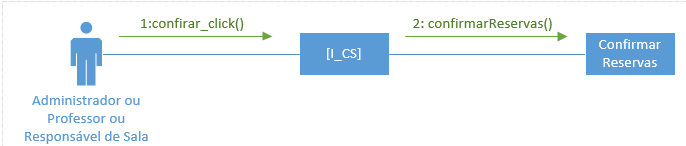
Adicionar Reservas



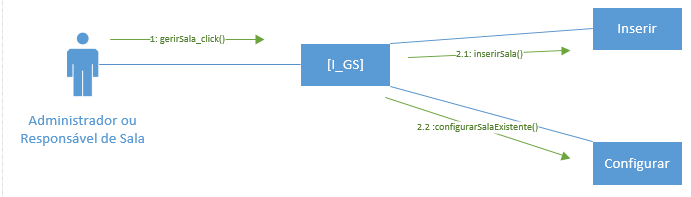
Excluir Reservas



Confirmar a Reserva



Gerir Sala



Gerir Utilizadores

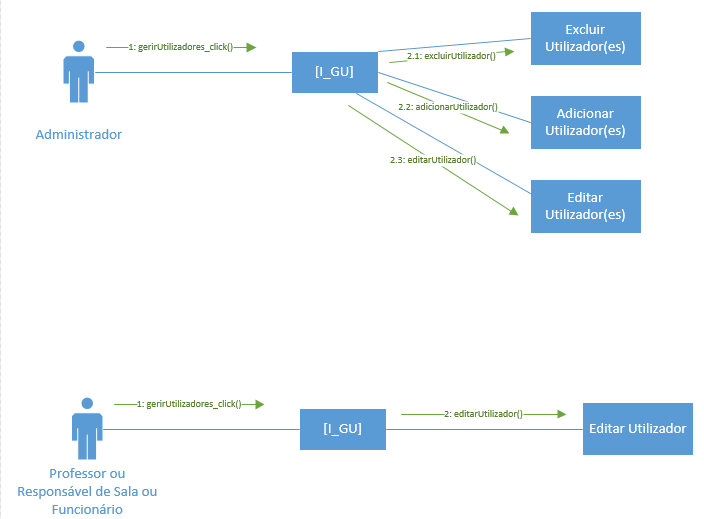


Diagrama de Estados

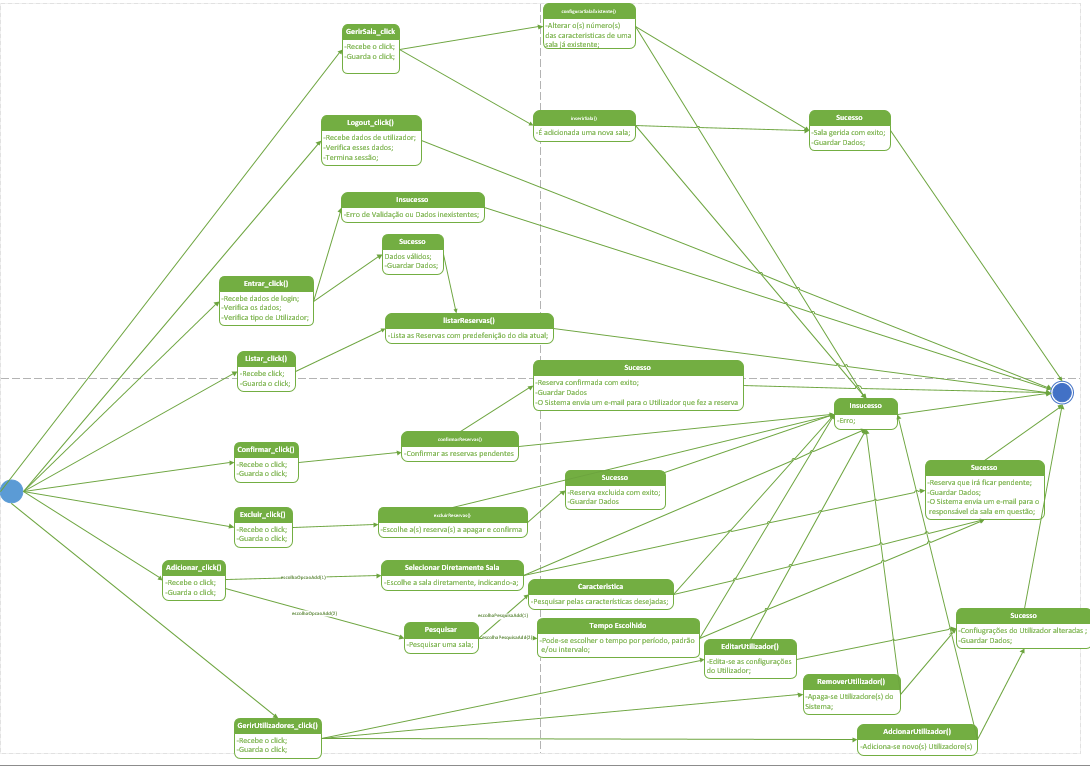
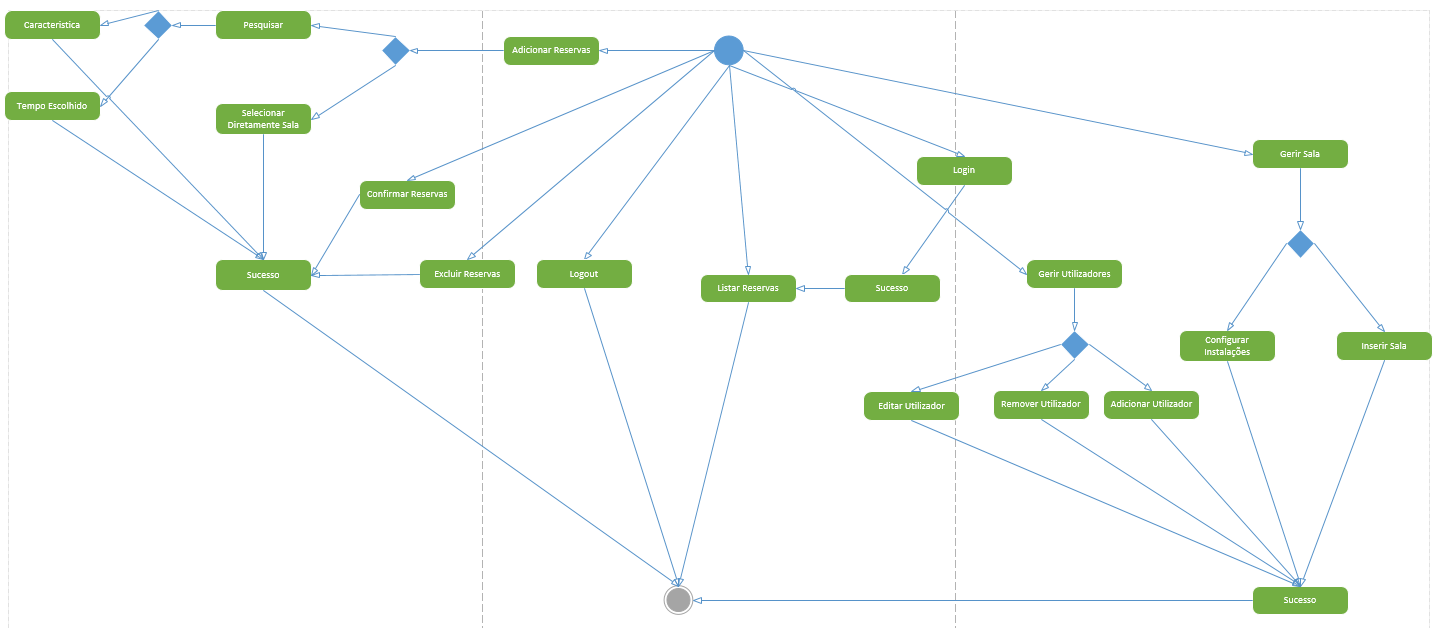


Diagrama de Atividades



Implementação dos Subprogramas Armazenados

Funcionalidade 1

Esta funcionalidade tem como objetivo retornar a lista de salas existentes e carateristicas, o código de criação é o seguinte:

CREATE OR REPLACE VIEW getSalas AS

SELECT SALAS.ID\_SALA, SALAS.LOCALIZACAO, SALAS.RESPONSAVEL, UTILIZADORES.NOME, CARATERISTICAS.ID\_CARATERISTICA, CARATERISTICAS.NOME\_CARATERISTICA, CARATERISTICAS.SOFTWARE, SALAS\_CARATERISTICAS.VALOR

FROM SALAS

INNER JOIN SALAS\_CARATERISTICAS ON SALAS.ID\_SALA = SALAS\_CARATERISTICAS.SALA

INNER JOIN CARATERISTICAS ON CARATERISTICAS.ID\_CARATERISTICA = SALAS\_CARATERISTICAS.CARATERISTICA

INNER JOIN UTILIZADORES ON SALAS.RESPONSAVEL = UTILIZADORES.ID\_UTILIZADOR

ORDER BY SALAS.LOCALIZACAO;

Funcionalidade 2

Esta funcionalidade tem como objetivo listar salas que cumpram os requesitos, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getSalas\_Carateristicas(v\_carateristicas IN tabc\_valor, lista OUT SYS\_REFCURSOR)

IS BEGIN

OPEN lista FOR SELECT getSalas.ID\_SALA

FROM getSalas, TABLE(v\_carateristicas) v\_carateristicas

WHERE getSalas.NOME\_CARATERISTICA = v\_carateristicas.CARATERISTICA

AND getSalas.VALOR >= v\_carateristicas.VALOR

GROUP BY getSalas.ID\_SALA

ORDER BY getSalas.ID\_SALA;

END;

/

Funcionalidade 3

Esta funcionalidade tem como objetivo retornar uma sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getSala(v\_id IN NUMBER, lista OUT SYS\_REFCURSOR)

IS BEGIN

OPEN lista FOR SELECT \* FROM SALAS

WHERE ID\_SALA = v\_id

ORDER BY ID\_SALA;

END;

/

Funcionalidade 4

Esta funcionalidade tem como objetivo listar todas as salas, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getAllSalas(lista OUT SYS\_REFCURSOR)

IS BEGIN

OPEN lista FOR SELECT \* FROM SALAS

ORDER BY ID\_SALA;

END;

/

Funcionalidade 5

Esta funcionalidade tem como objetivo adicionar uma sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE addEmptySala(NOME IN VARCHAR2, RESPONSAVEL IN NUMBER)

IS BEGIN

INSERT INTO SALAS VALUES ((SELECT COUNT(\*) + 1 FROM SALAS), NOME, RESPONSAVEL);

END;

/

Funcionalidade 6

Esta funcionalidade tem como objetivo adicionar uma sala e respetivas carateristicas, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE addSala(LOCALIZACAO VARCHAR2, RESPONSAVEL NUMBER, v\_carateristicas IN tabc\_valor)

IS l\_index NUMBER;

id NUMBER;

BEGIN

INSERT INTO SALAS VALUES((SELECT COUNT(\*) + 1 FROM SALAS), LOCALIZACAO, RESPONSAVEL)

RETURNING ID\_SALA INTO id;

l\_index := v\_carateristicas.FIRST;

WHILE (l\_index IS NOT NULL) LOOP

addCar(id, v\_carateristicas(l\_index).CARATERISTICA, v\_carateristicas(l\_index).VALOR);

l\_index := v\_carateristicas.NEXT(l\_index);

END LOOP;

END addSala;

/

Funcionalidade 7

Esta funcionalidade tem como objetivo apagar uma sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE popSala(v\_idSala IN NUMBER)

IS BEGIN

DELETE FROM SALAS WHERE ID\_SALA = v\_idSala;

END popSala;

/

Funcionalidade 8

Esta funcionalidade tem como objetivo editar uma sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE editSala(LOCALIZACAO VARCHAR2, RESPONSAVEL NUMBER, v\_idSala IN NUMBER, v\_carateristicas IN tabc\_valor, lista OUT SYS\_REFCURSOR)

IS BEGIN

addSala(LOCALIZACAO, RESPONSAVEL, v\_carateristicas);

UPDATE SALAS SET

ID\_SALA = v\_idSala

WHERE ID\_SALA = (SELECT COUNT(\*) + 1 FROM SALAS);

END editSala;

/

Funcionalidade 9

Esta funcionalidade tem como objetivo retornar o horario de uma certa sala numa certa data, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getSalas\_horario(v\_sala IN VARCHAR2, v\_dia IN DATE, horarios OUT SYS\_REFCURSOR)

IS BEGIN

OPEN horarios FOR

SELECT RESERVAS.ID\_RESERVA, RESERVAS.HORA\_INICIO, RESERVAS.HORA\_FIM, RESERVAS.TURMA, RESERVAS.DISCIPLINA, RESERVAS.DOCENTE, RESERVAS.PENDENTE, RESERVAS.DESCRICAO

FROM RESERVAS, SALAS

WHERE SALAS.LOCALIZACAO = v\_sala AND RESERVAS.SALA = SALAS.ID\_SALA AND

(DIA = v\_dia OR

DIA\_SEMANA = TO\_CHAR(v\_dia, 'DY'));--NAO FUNCIONA

END getSalas\_horario;

/

Funcionalidade 10

Esta funcionalidade tem como objetivo dizer se a sala esta disponível num certo dia e hora, o código de criação é o seguinte:

CREATE OR REPLACE FUNCTION salaDisponivel(v\_sala IN VARCHAR2, v\_dia IN DATE, v\_inicio IN NUMBER, v\_fim IN NUMBER)

RETURN NUMBER IS

valor NUMBER := 0;

CURSOR horarios (v\_sala IN VARCHAR2, v\_dia IN DATE) IS

SELECT RESERVAS.ID\_RESERVA, RESERVAS.HORA\_INICIO, RESERVAS.HORA\_FIM, RESERVAS.TURMA, RESERVAS.DISCIPLINA, RESERVAS.DOCENTE, RESERVAS.PENDENTE, RESERVAS.DESCRICAO

FROM RESERVAS, SALAS

WHERE SALAS.LOCALIZACAO = v\_sala AND RESERVAS.SALA = SALAS.ID\_SALA AND

(DIA = v\_dia OR

DIA\_SEMANA = TO\_CHAR(v\_dia, 'DY'));

BEGIN

FOR horario in horarios(v\_sala, v\_dia) LOOP

IF(horario.HORA\_INICIO = v\_inicio) THEN

valor := 1;

END IF;

IF(valor = 1) THEN

IF(horario.PENDENTE = 0) THEN

RETURN 0;

END IF;

IF(horario.HORA\_FIM = v\_fim) THEN

valor := 0;

END IF;

END IF;

END LOOP;

Return 1;

END salaDisponivel;

/

Funcionalidade 11

Esta funcionalidade tem como objetivo mostrar pedidos de reserva de um docente, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getPedido(v\_id IN VARCHAR2, pedidos OUT SYS\_REFCURSOR)

IS BEGIN

OPEN pedidos FOR

SELECT RESERVAS.ID\_RESERVA, RESERVAS.SALA, SALAS.LOCALIZACAO, RESERVAS.HORA\_INICIO, RESERVAS.HORA\_FIM, RESERVAS.DIA\_SEMANA, RESERVAS.TURMA, RESERVAS.DISCIPLINA, RESERVAS.DOCENTE, RESERVAS.DIA, RESERVAS.PENDENTE, RESERVAS.DESCRICAO

FROM RESERVAS INNER JOIN SALAS ON SALAS.ID\_SALA = RESERVAS.SALA

WHERE RESERVAS.DOCENTE = (SELECT ID\_UTILIZADOR FROM UTILIZADORES WHERE ID\_UTILIZADOR = v\_id);

END getPedido;

/

Funcionalidade 12

Esta funcionalidade tem como objetivo mostrar pedidos de reserva pendentes de um responsavel de sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getPedidoPendente(v\_id IN VARCHAR2, pedidos OUT SYS\_REFCURSOR)

IS BEGIN

OPEN pedidos FOR

SELECT RESERVAS.ID\_RESERVA, RESERVAS.SALA, SALAS.LOCALIZACAO, RESERVAS.HORA\_INICIO, RESERVAS.HORA\_FIM, RESERVAS.DIA\_SEMANA, RESERVAS.TURMA, RESERVAS.DISCIPLINA, RESERVAS.DOCENTE, RESERVAS.DIA, RESERVAS.PENDENTE, RESERVAS.DESCRICAO

FROM RESERVAS INNER JOIN SALAS ON SALAS.ID\_SALA = RESERVAS.SALA

WHERE SALAS.RESPONSAVEL = (SELECT ID\_UTILIZADOR FROM UTILIZADORES WHERE ID\_UTILIZADOR = v\_id);

END getPedidoPendente;

/

Funcionalidade 12

Esta funcionalidade tem como objetivo aceitar um pedido de reserva, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE aceitarPedido(v\_pedido IN NUMBER)

IS BEGIN

UPDATE RESERVAS SET RESERVAS.PENDENTE = '0'

WHERE RESERVAS.ID\_RESERVA = v\_pedido;

END aceitarPedido;

/

Funcionalidade 13 e 14

Estas funcionalidades têm como objetivo inserir pedidos de reserva, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE addPedido2(SALA IN VARCHAR2, HORA\_INICIO IN NUMBER, HORA\_FIM NUMBER,

DIA\_SEMANA IN VARCHAR2, TURMA IN VARCHAR2, DISCIPLINA IN VARCHAR2, DOCENTE IN NUMBER, DIA IN DATE, PENDENTE IN CHAR, DESCRICAO IN VARCHAR2)

IS BEGIN

INSERT INTO RESERVAS VALUES((SELECT COUNT(\*) FROM RESERVAS),

(SELECT ID\_SALA FROM Salas WHERE LOCALIZACAO = SALA),

HORA\_INICIO, HORA\_FIM, DIA\_SEMANA, TURMA, DISCIPLINA, DOCENTE, DIA, PENDENTE, DESCRICAO);

END;

/

CREATE OR REPLACE PROCEDURE addPedido(SALA IN NUMBER, HORA\_INICIO IN NUMBER, HORA\_FIM NUMBER,

DIA\_SEMANA IN VARCHAR2, TURMA IN VARCHAR2, DISCIPLINA IN VARCHAR2, DOCENTE IN NUMBER, DIA IN DATE, PENDENTE IN CHAR, DESCRICAO IN VARCHAR2)

IS BEGIN

INSERT INTO RESERVAS VALUES((SELECT COUNT(\*) FROM RESERVAS),

SALA, HORA\_INICIO, HORA\_FIM, DIA\_SEMANA, TURMA, DISCIPLINA, DOCENTE, DIA, PENDENTE, DESCRICAO);

END addPedido;

/

Funcionalidade 15

Esta funcionalidade tem como objetivo apagar pedidos de reserva pendentes, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE popPedidoPendente(v\_pedido IN NUMBER)

IS BEGIN

DELETE FROM RESERVAS

WHERE RESERVAS.ID\_RESERVA = v\_pedido;

END popPedidoPendente;

/

Funcionalidade 16

Esta funcionalidade tem como objetivo editar um pedido de reserva pendente, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE editPedidoPendente(v\_pedido IN NUMBER, v\_SALA IN NUMBER, v\_HORA\_INICIO IN NUMBER, v\_HORA\_FIM IN NUMBER,

v\_DIA\_SEMANA IN VARCHAR2, v\_TURMA IN VARCHAR2, v\_DISCIPLINA IN VARCHAR2, v\_DOCENTE IN NUMBER, v\_DIA IN DATE, v\_DESCRICAO IN VARCHAR2)

IS BEGIN

UPDATE RESERVAS SET

SALA = v\_SALA,

HORA\_INICIO = v\_HORA\_INICIO,

HORA\_FIM = v\_HORA\_FIM,

DIA\_SEMANA = v\_DIA\_SEMANA,

TURMA = v\_TURMA,

DISCIPLINA = v\_DISCIPLINA,

DOCENTE = v\_DOCENTE,

DIA = v\_DIA,

DESCRICAO = v\_DESCRICAO

WHERE ID\_RESERVA = v\_pedido;

END editPedidoPendente;

/

Funcionalidade 17

Esta funcionalidade tem como objetivo retorna o utilizador logado, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getUser(v\_email IN VARCHAR2, v\_pass IN VARCHAR2, v\_lista OUT SYS\_REFCURSOR) IS

BEGIN

OPEN v\_lista FOR

SELECT \* FROM Utilizadores WHERE(EMAIL = v\_email AND PALAVRA\_PASSE = v\_pass);

END;

/

Funcionalidade 18

Esta funcionalidade tem como objetivo adicionar Utilizadores, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE AddUser(V\_NOME IN VARCHAR2, V\_PALAVRA\_PASSE IN VARCHAR2, V\_EMAIL IN VARCHAR2, V\_TIPO\_USER IN CHAR)

IS BEGIN

INSERT INTO Utilizadores VALUES((SELECT COUNT(\*) + 1 FROM UTILIZADORES) , V\_NOME, V\_PALAVRA\_PASSE, V\_EMAIL, V\_TIPO\_USER);

END;

/

Funcionalidade 19

Esta funcionalidade tem como objetivo apagar Utilizadores, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE PopUser(v\_id IN NUMBER)

IS BEGIN

DELETE FROM Utilizadores WHERE ID\_UTILIZADOR = v\_id;

END;

/

Funcionalidade 20

Esta funcionalidade tem como objetivo apagar professores, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE PopProf(v\_id IN NUMBER)

IS BEGIN

DELETE FROM RESERVAS WHERE DOCENTE = v\_id;

DELETE FROM Utilizadores WHERE ID\_UTILIZADOR = v\_id;

END;

/

Funcionalidade 21

Esta funcionalidade tem como objetivo apagar responsáveis de sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE PopResp(v\_id IN NUMBER, v\_new IN NUMBER)

IS BEGIN

UPDATE SALAS SET RESPONSAVEL = v\_new WHERE RESPONSAVEL = v\_id;

DELETE FROM RESERVAS WHERE DOCENTE = v\_id;

DELETE FROM Utilizadores WHERE ID\_UTILIZADOR = v\_id;

END;

/

Funcionalidade 22

Esta funcionalidade tem como objetivo editar utilizadores, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE EditUser(v\_id IN NUMBER, V\_NOME IN VARCHAR2, V\_PALAVRA\_PASSE IN VARCHAR2, V\_EMAIL IN VARCHAR2, V\_TIPO\_USER IN CHAR)

IS BEGIN

UPDATE Utilizadores SET

NOME = V\_NOME,

PALAVRA\_PASSE = V\_PALAVRA\_PASSE,

EMAIL = V\_EMAIL,

TIPO\_USER = V\_TIPO\_USER

WHERE ID\_UTILIZADOR = v\_id;

END;

/

Funcionalidade 23

Esta funcionalidade tem como objetivo adicionar uma sala a um utilizador, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE addSala\_User(v\_user IN NUMBER, v\_sala IN NUMBER)

IS BEGIN

UPDATE SALAS SET RESPONSAVEL = v\_user

WHERE ID\_SALA = v\_sala;

END;/

Funcionalidade 24

Esta funcionalidade tem como objetivo listar todos os utilizadores, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE listaUsers(v\_cursor OUT SYS\_REFCURSOR)

IS BEGIN

OPEN v\_cursor FOR

SELECT \* FROM UTILIZADORES ORDER BY ID\_UTILIZADOR;

END;

/

Funcionalidade 25

Esta funcionalidade tem como objetivo listar todos os responsáveis de sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE listaResponsavel(v\_cursor OUT SYS\_REFCURSOR)

IS BEGIN

OPEN v\_cursor FOR

SELECT \* FROM UTILIZADORES

WHERE TIPO\_USER = 'R' ORDER BY ID\_UTILIZADOR;

END;

/

Funcionalidade 26

Esta funcionalidade tem como objetivo listar adicionar uma nova caraterística, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE addNewCar(NOME IN VARCHAR2, e\_soft IN CHAR)

IS BEGIN

INSERT INTO CARATERISTICAS VALUES((SELECT COUNT(\*) + 1 FROM CARATERISTICAS), NOME, e\_soft);

END;

/

Funcionalidade 27

Esta funcionalidade tem como objetivo apagar uma caraterística, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE popCar(id IN NUMBER)

IS BEGIN

DELETE FROM CARATERISTICAS WHERE ID\_CARATERISTICA = id;

END;

/

Funcionalidade 28

Esta funcionalidade tem como objetivo editar uma caraterística, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE editCar(id IN NUMBER, v\_NOME IN VARCHAR2, e\_soft IN CHAR)

IS BEGIN

UPDATE CARATERISTICAS SET

NOME\_CARATERISTICA = v\_NOME,

SOFTWARE = e\_soft

WHERE ID\_CARATERISTICA = id;

END;

/

Funcionalidade 29, 30 e 31

Esta funcionalidade tem como objetivo adicionar uma caraterística a uma sala, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE addCar(sala IN NUMBER, id IN VARCHAR2, valor IN NUMBER)

IS BEGIN

INSERT INTO Salas\_Carateristicas VALUES(sala,

(SELECT ID\_CARATERISTICA FROM CARATERISTICAS WHERE NOME\_CARATERISTICA = id), valor);

END;

/

CREATE OR REPLACE PROCEDURE addCarat(sala IN VARCHAR2, id IN NUMBER, valor IN NUMBER)

IS BEGIN

INSERT INTO Salas\_Carateristicas VALUES(

(SELECT ID\_SALA FROM Salas WHERE Salas.LOCALIZACAO = sala), id, valor);

END;

/

CREATE OR REPLACE PROCEDURE addCarSala(v\_sala IN VARCHAR2, id\_car IN VARCHAR2, valor IN NUMBER)

IS BEGIN

INSERT INTO Salas\_Carateristicas VALUES(

(SELECT ID\_SALA FROM SALAS WHERE LOCALIZACAO = v\_sala),

(SELECT ID\_CARATERISTICA FROM CARATERISTICAS WHERE NOME\_CARATERISTICA = id\_car),

valor);

END;

/

Funcionalidade 32

Esta funcionalidade tem como objetivo fazer o login, o código de criação é o seguinte:

CREATE OR REPLACE FUNCTION DoLogin(v\_email IN VARCHAR2, v\_pass IN VARCHAR2)

RETURN number IS cnumber number;

BEGIN

SELECT COUNT(\*) INTO cnumber FROM Utilizadores WHERE(EMAIL = v\_email AND PALAVRA\_PASSE = v\_pass);

RETURN cnumber;

END;

/

Funcionalidade 33

Esta funcionalidade tem como objetivo obter a característica pedida, o código de criação é o seguinte:

CREATE OR REPLACE PROCEDURE getCarateristicas(v\_cursor OUT SYS\_REFCURSOR)

IS BEGIN

OPEN v\_cursor FOR

SELECT \* FROM CARATERISTICAS ORDER BY ID\_CARATERISTICA;

END;

/

Funcionalidade 34

Esta funcionalidade tem como objectivo definir o responsável de cada sala, o código de criação é o seguinte:

CREATE OR REPLACE TRIGGER E\_RESPONSAVEL

BEFORE INSERT OR UPDATE ON SALAS

FOR EACH ROW

DECLARE V\_TIPO\_USER CHAR(1);

user\_invalido EXCEPTION;

BEGIN

SELECT TIPO\_USER INTO V\_TIPO\_USER FROM UTILIZADORES WHERE ID\_UTILIZADOR = :NEW.RESPONSAVEL;

IF V\_TIPO\_USER <> 'R' THEN

RAISE user\_invalido;

END IF;

EXCEPTION

WHEN user\_invalido THEN

DBMS\_OUTPUT.PUT\_LINE('Utilizador invalido');

END GERA\_ID\_RESERVA;

/

Conclusões

Após a realização deste projeto, acreditamos que este projeto criado dá uma resposta ao problema imposto. Adquirimos uma perceção diferente da necessidade de utilizar diagramas UML para o planeamento do projeto, vimos que esses diagramas são uma salvaguarda para o processo de desenvolvimento do sistema. Apenas sentimos algumas dificuldades em conectar o web site à base de dados e em algumas funcionalidades do sistema. Por fim concluímos que com a elaboração deste sistema contribui bastante para a nossa formação como alunos, porque nos permitiu aplicar e desenvolver o conhecimento adquirido nas unidades curriculares lecionadas, tal como adquirir novos conhecimentos e práticas sobre os mesmos.

O nosso sistema está funcional, onde os utilizadores podem aceder ao sistema com as devidas credenciais, credenciais essas que diferem consoante o tipo de utilizador. Os utilizadores, consoante o tipo, podem realizar as funções disponíveis no sistema, permitindo fazer uma gestão apropriada da ocupação das salas, cumprindo assim o objetivo deste projeto.

Bibliografia

SAMARTINHO, João; NASCIMENTO, João – Diagramas de Casos de Utilização. Instituto Politécnico de Santarém – Escola Superior de Tecnologias e Gestão de Santarém, 18/10/2016

SAMARTINHO, João; NASCIMENTO, João – Diagramas de Classes. Instituto Politécnico de Santarém – Escola Superior de Tecnologias e Gestão de Santarém, 15/11/2016

SAMARTINHO, João, NASCIMENTO, João – Diagramas de Sequências. Instituto Politécnico de Santarém – Escola Superior de Tecnologias e Gestão de Santarém, 17/11/2016

SAMARTINHO, João, NASCIMENTO, João – Diagramas de Comunicação e de Estado. Instituto Politécnico de Santarém – Escola Superior de Tecnologias e Gestão de Santarém, 28/11/2016

SAMARTINHO, João; NASCIMENTO, João – Diagramas de Atividade. Instituto Politécnico de Santarém – Escola Superior de Tecnologias e Gestão de Santarém, 12/12/2016