

Desenvolvimento de protótipo de uma aplicação baseada em *Java Application Server*

Myclinic - Cuidados médicos

26 de novembro de 2019

Módulo: Programação em Java
Curso: Aprender Java.Net
Da autoria de: Alberto Mões
Formador: Carlos Jorge A. Costa

Índice

Lista de abreviaturas e siglas	3
Introdução	4
Problema	5
Equipamentos e Tecnologias	5
Capítulo 1 Procedimento.....	6
1.1. Engenharia de requisitos	6
1.2. Conceção	7
1.2.1. Atores.....	7
1.2.2. Class e Package	8
1.3. Implementação.....	9
1.3.1 Diagrama de implementação	9
1.3.2. Concretização.....	9
1.3.2.1. Base de dados.....	9
1.3.2.2. Sistema de Login	9
1.3.2.3. Encriptação da palavra-passe	10
1.3.2.4. Menu	10
1.3.2.5. Novos dados	10
1.3.2.6. Editar e eliminar dados	10
1.3.2.7. Prescrição	10
1.3.2.8. Gestão de utilizadores.....	10
Testes	11
Trabalho Futuro	12
Conclusão	12
Biografia	13
Anexos.....	14

Lista de abreviaturas e siglas

CDI	Contexts and Dependency Injection
IDE	integrated development environment
IU	User Interface – Interface de utilizador
JDBC	Java Database Connectivity
JSF	JavaServer Faces
JDK	Java Development Kit
MVC	Model-View-Controller – padrão de arquitetura de software formulado na década de 1970
MySQL	open-source relational database management system
Login	processo para acessar um sistema informático restrito feita através da autenticação
Logout	processo para deixar cair o acesso a um sistema informático

Introdução

O presente relatório representa a experiência no módulo “*Programar em Java*” do curso Aprender Java.Net da Iniciativa Portugal INCoDe.2030.

Consistiu em aplicar os conhecimentos e técnicas aprendidas no desenvolvimento de um protótipo de uma aplicação baseada em “*Java Application Server*”.

O estudo realizado é fictício, mas serve para aplicar diversas tecnologias, e adaptá-lo a um caso real.

Problema

Desenvolvimento de uma aplicação web em três camadas (MVC) para gestão de uma clínica de cuidados médicos.

Equipamentos e Tecnologias

- ✓ IDE: Eclipse IDE for Enterprise Java Developers «2019-09 R (4.13.0)»
- ✓ Maven – M2Eclipse «1.13» - Gerenciador e construtor de projetos
- ✓ JavaServer Faces (JSF) «2.4.0» - desenho de IU (interface de utilizador)
- ✓ PrimeFaces «7.0» – IU library
- ✓ BootsFaces «1.4.0» – IU library – O Bootstrap para JSF
- ✓ JDBC «4.2», e conector MySQL «mysql-connector-java 8.0.18» – Ligação à base de dados
- ✓ Hibernate e Java Persistence API (JPA) «2.0»
- ✓ Contexts and Dependency Injection (CDI) «2.0»
- ✓ WildFly «18.0» Servidor de aplicação de código fonte aberto
- ✓ JDK 12.0.2
- ✓ Equipamento com Windows 10 Pro @2018

Capítulo 1 Procedimento

Seguem as fases de desenvolvimento de software

1.1. Engenharia de requisitos

O sistema tem de contemplar o registo de consultas, pacientes, médicos e utilizadores.

Na consulta deve constar a data e hora, o paciente e o médico. Implementar a possibilidade de cancelamento (sobe pedido aos administrativos). O registo da prescrição pelo médico, só por este. A marcação e alterações das consultas são realizadas pelos administrativos e médico.

Dados, geridos pelos administrativos:

- pacientes – nome, morada, NIF, e-mail e contacto;
- médicos – nome, morada, NIF, e-mail, contacto e especialidade;

Dados, geridos pelos administradores:

- utilizadores – nome-utilizador, senha e nível de acesso, onde, usamos o e-mail ou o NIF para associar o utilizador ao paciente e para associar ao médico, e o nível de acesso, podem ser de paciente, de médico, de administrativo e de administrador.

Áreas:

- paciente – consta a lista das suas consultas e respetiva prescrição.
- médico – consta a lista dos pacientes, as suas consultas, e marcar novas consultas para si.
- administrativos – lista das consultas, médicos e pacientes, só não podem eliminar dados.
- administrador – deve criar acessos aos atores do sistema, acesso total com poder de eliminar dados. Não podem editar e passar prescrições. E só atua no sistema se solicitado (ator passivo).

1.2. Conceção

1.2.1. Atores

Diagrama Casos de Uso



Figura 1 - Diagrama casos de uso

1.2.2. Class e Package

Diagrama de Classes

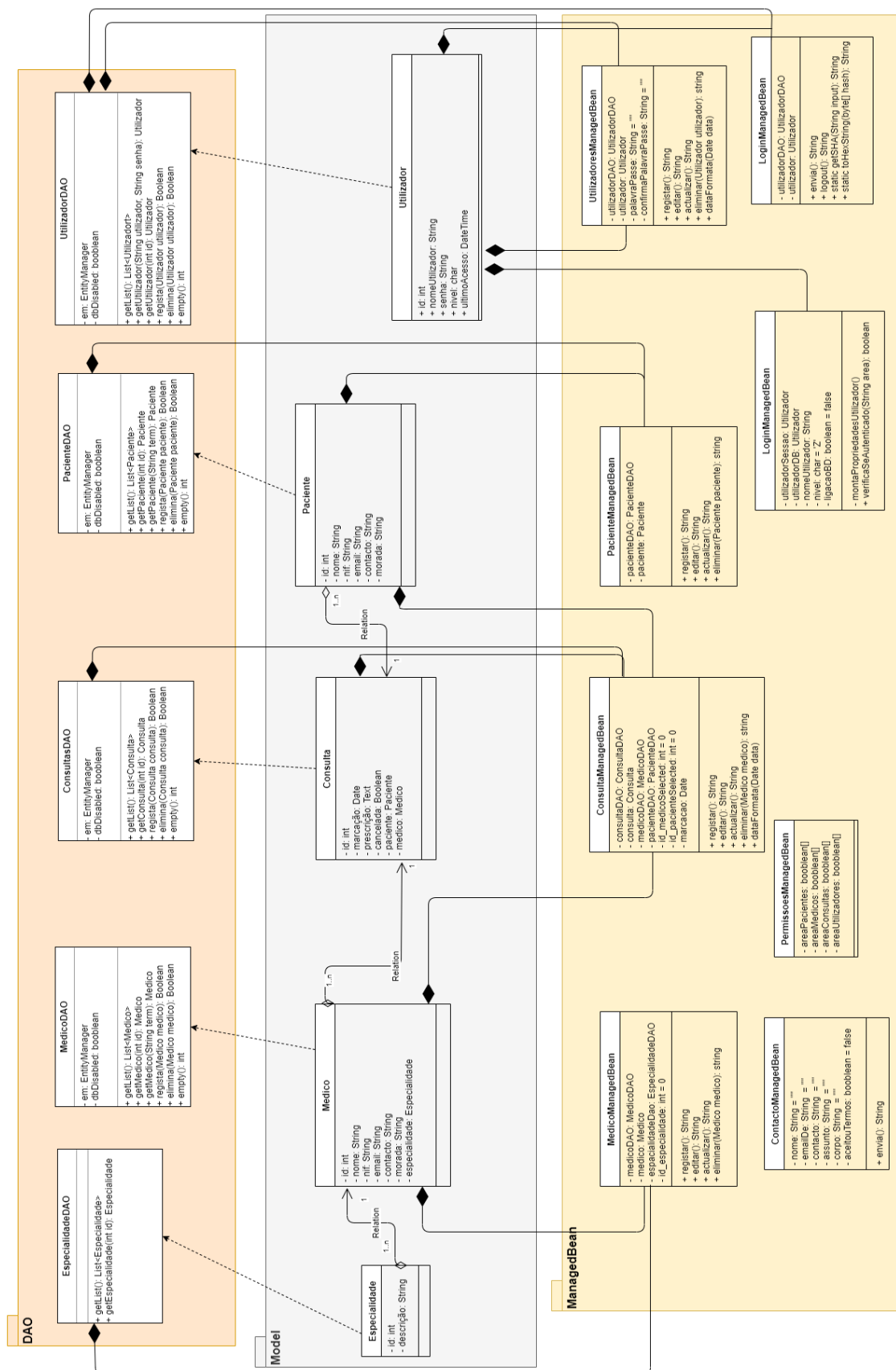


Figura 2 - Diagrama de classes e package

1.3. Implementação

1.3.1 Diagrama de implementação

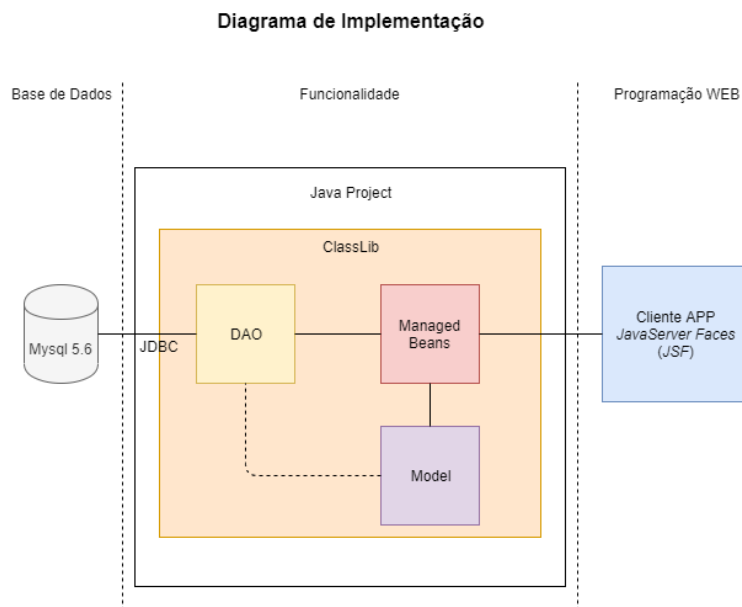


Figura 3 - Diagrama de Implementação

1.3.2. Concretização

1.3.2.1. Base de dados

Com auxílio das tecnologias Hibernate e JPA temos a abstração para construir a nossa base de dados. No ficheiro persistence.xml¹ são indicadas as classes a mapear para a base de dados, elas, tecnologias, criam as entidades, atributos e as relações². Criamos a base de dados “myclinic” no mysql e referenciamos no ficheiro persistence.xml.

1.3.2.2. Sistema de Login

O sistema de login³ implementado tem por objetivo fornecer acesso seguro e controlado aos utilizadores da aplicação. Quando o utilizador efetua com sucesso o login é guardado a data e hora local na base de dados, e é direcionado para a área de trabalho, inicializa uma nova sessão. O utilizador ao clicar em “Sair” no menu faz logout, se termina a sua sessão e é redirecionado para a página inicial (login).

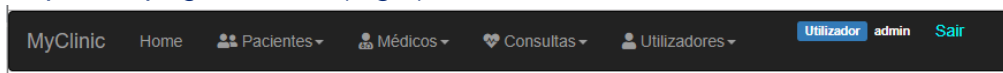


Figura 4 – menu do administrador

¹ Anexo A – ficheiro persistence.xml

² Anexo B – diagrama ER

³ Anexo C – página de Login

1.3.2.3. Encriptação da palavra-passe

A classe “SHA” criada para “*secure hash algorithm*” que pega na “*string*” da palavra-passe inserida e encripta para SHA-256⁴.

1.3.2.4. Menu

Adaptação do menu aos diferentes níveis, define as diferentes áreas de trabalho. Criada a classe “*MangedBear*” (*PermissoesManagedBear*⁵) que trata das permissões, e junto com o poder da propriedade “*rendered*” dos elementos do “*BootsFaces*”, a adaptação do menu⁶ é simples.

1.3.2.5. Novos dados

Utilizado o ponto anterior, fica bem definido quem pode inserir novos dados. A opção “*Inserir*” do menu só é disponibilizada a quem o nível de acesso permite fazê-lo.

1.3.2.6. Editar e eliminar dados

Usa a classe *PermissoesManagedBean*⁵ e a propriedade “*rendered*” dos elementos do “*PrimeFaces*”. Exemplo na lista de consultas⁷ (linha 43 e linha 48).

1.3.2.7. Prescrição

A classe *PermissoesManagedBean*⁵ define mais uma vez quem pode passar/editar as prescrições. Vemos que só o nível médico tem permissão para passar/editar prescrição⁸ (linha 42).

1.3.2.8. Gestão de utilizadores

A gestão dos utilizadores é feita pelos administradores do sistema. São eles que podem associar o utilizador a um paciente ou a um médico, usando o NIF ou o e-mail como referência e definir o nível de acesso (Paciente, Médico, Administrativo e Administrador)⁹.

⁴ Anexo D – classe *SHA*

⁵ Anexo E – parte da classe *PermissoesManagedBean*

⁶ Anexo F – ficheiro *menu.xhtml*

⁷ Anexo G – ficheiro *index.xhtml* da consulta

⁸ Anexo H – ficheiro *editar.xhtml* da consulta

⁹ Anexo I – pagina inserir novo utilizador

Testes

Teste	Dados usados	Estado	Resultado
Autenticação dos utilizadores no sistema	As credenciais pessoais	Sucesso	Se se verifica a autenticidade das entradas, a página é redirecionada para a área pessoal, caso contrário exibe mensagem de credenciais não válidas.
Fazer <i>logout</i> do sistema	-	Sucesso	Verifica-se seja qual for o nível que esteja em sessão, ao clicar em “Sair” disponível no Menu, é terminada a sessão e a página direcionada para a página inicial (<i>login</i>).
Adaptação do menu de navegação ao nível do utilizador	Nível do utilizador	Sucesso	Verifica-se que o menu se adapta a cada nível.
O botão eliminar só para os administradores	Nível administrador	Sucesso	Verifica-se que o botão que dá a opção eliminar só aparece aos administradores.
Os administradores têm acesso a todas a listagens e botões, exceto passar e editar as prescrições	Nível administrador	Sucesso	Verifica-se que as listagens estão disponíveis e os botões editar, eliminar e cancelar aparecem aos administradores. E que alterar ou inserir uma prescrição não está disponível.
O botão editar paciente e botão editar médico não deve aparecer aos próprios.	Nível paciente e nível médico	Sucesso	Verifica-se que o botão que dá a opção editar não aparece ao nível paciente nem ao nível médico.
A opção de passar e editar prescrição só deve aparecer aos médicos	Nível médico	Sucesso	Verifica-se que a opção de escrever na caixa da prescrição só aparece para o nível médico.
Se o utilizador for médico ou paciente a listagem de consultas só devem constar as consultas destes	Nível médico e nível paciente	Sucesso	O paciente autenticado e pedir a listagem das consultas só contam as dele, verificando-se o mesmo para o médico.
O botão de editar consulta só deve aparecer aos administrativos e ao médico	Nível médico e nível administrativo	Sucesso	Verifica-se que botão editar consulta só aparece ao nível administrativo e ao médico da consulta.
O botão cancelar consultas só deve aparecer aos administrativos	Nível administrativo	Sucesso	Verifica-se que o botão cancelar só é disponibilizado ao nível administrativo
O CRUD realiza-se como planeado	Novos, editar, eliminar e listar	Sucesso	Verifica-se que as regras definidas correm como esperado.
Não pode ser possível marcar consulta para uma data inferior à data e hora atual	Nível médico, nível administrativo e nível administrador	Sucesso	Verifica-se que ao inserir uma data e hora anterior à data e hora atual, é devolvido uma mensagem de erro.

Trabalho Futuro

Dado que este projeto foi desenvolvido sobre um caso fictício existem várias questões que podem ser melhoradas e funcionalidades que poderiam ajudar o utilizador.

Uma das melhorias reside no facto do sistema vir a implementar a anexação de ficheiros.

Outra melhoria seria os próprios pacientes e médicos criarem e gerirem as suas contas de utilizador. Deixando assim livre os administradores desta tarefa.

Casos de estudo com o objetivo de melhorar o produto final.

Conclusão

Com este projeto foi possível colocar em prática grande parte do conteúdo teórico aprendido em sala de aula.

A interpretação da análise de requisitos, espírito crítico e proactivo foram colocados à prova durante o projeto, obrigando a necessidade de encontrar resolução para problemas encontrados durante a implementação da aplicação.

Para além das capacidades adquiridas no projeto foram obtidas outras competências como *soft-skills*.

Para finalizar é possível afirmar que o projeto foi realizado com sucesso não só em termos de trabalho concluído bem como nos aspetos formativos.

Biografia

[1] Documentação do módulo e restante curso. Disponível em: < <https://qualifica.ipv.pt/>>. Último acesso em: 25 de novembro de 2019.

[2] Bootsfaces. Disponível em: <<https://showcase.bootsfaces.net/>>. Último acesso em: 18 de novembro de 2019.

[3] Primefaces. Disponível em: <<https://www.primefaces.org/showcase/>>. Último acesso em: 18 de novembro de 2019.

[4] JavaServer™ Faces (JSF). Disponível em <<http://www.java-serverfaces.org/>>. Último acesso em: 17 de novembro de 2019.

[5] Hibernate. Disponível em <<https://hibernate.org/>>. Último acesso em: 17 de novembro de 2019.

[6] Java Persistence API (JPA). Disponível em <<https://www.oracle.com/technetwork/java/javaee/tech/persistence-jsp-140049.html>>. Último acesso em: 17 de novembro de 2019.

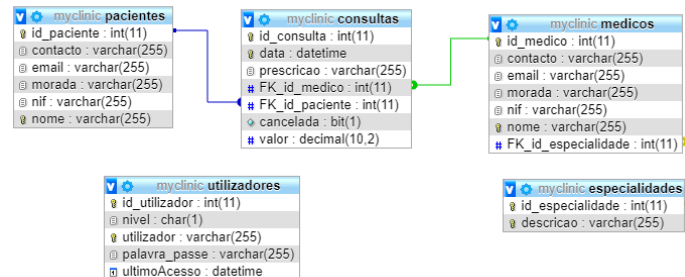
[7] Hibernate e JPA. Disponível em < <https://www.javaworld.com/article/3379043/what-is-jpa-introduction-to-the-java-persistence-api.html>>. Último acesso em: 17 de novembro de 2019.

[8] Contexts and Dependency Injection for the Java EE Platform. Disponível em < <https://docs.oracle.com/javaee/6/tutorial/doc/giwhb.html>>. Último acesso em: 17 de novembro de 2019.

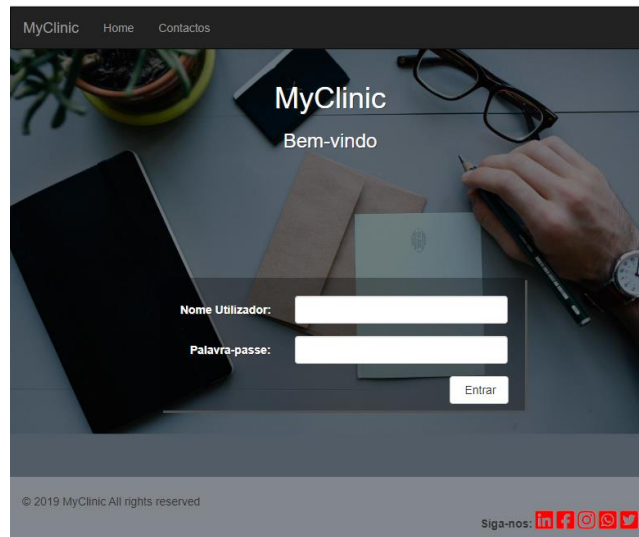
Anexos

```
1<?xml version="1.0" encoding="UTF-8"?>
2<persistence xmlns="http://java.sun.com/xml/ns/persistence"
3  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4  xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
5    http://java.sun.com/xml/ns/persistence/persistence_2_0.xsd"
6  version="2.0">
7  <persistence-unit name="myclinic">
8
9    <!-- provedor/implementacao do JPA -->
10    <provider>org.hibernate.ejb.HibernatePersistence</provider>
11
12    <!-- entidade mapeada -->
13    <class>pt.ipv.qualifica.myclinic.model.Utilizador</class>
14    <class>pt.ipv.qualifica.myclinic.model.Paciente</class>
15    <class>pt.ipv.qualifica.myclinic.model.Especialidade</class>
16    <class>pt.ipv.qualifica.myclinic.model.Medico</class>
17    <class>pt.ipv.qualifica.myclinic.model.Consulta</class>
18
19    <properties>
20
21      <!-- dados da conexao -->
22      <property name="javax.persistence.jdbc.driver"
23        value="com.mysql.cj.jdbc.Driver" />
24      <property name="javax.persistence.jdbc.url" value="jdbc:mysql://localhost/myclinic" />
25      <property name="javax.persistence.jdbc.user" value="root" />
26      <property name="javax.persistence.jdbc.password" value="" />
27
28      <!-- propriedades do hibernate -->
29      <property name="hibernate.dialect" value="org.hibernate.dialect.MySQL5InnoDBDialect" />
30      <property name="hibernate.show_sql" value="true" />
31      <property name="hibernate.format_sql" value="true" />
32
33      <!-- atualiza o banco, gera as entidades e relações se for preciso -->
34      <property name="hibernate.hbm2ddl.auto" value="update" />
35
36    </properties>
37  </persistence-unit>
38</persistence>
```

Anexo A – ficheiro persistence.xml



Anexo B – modelo ER da base de dados “myclinic”



Anexo C – página de Login

```

1  /*
4  package pt.ipv.qualifica.myclinic.util;
5
6  import java.math.BigInteger;
10
11  /**
12   * @author Programador
13   *
14   */
15  public class SHA {
16      /**
17       * Gera uma string codificada em SHA-256 da string passada
18       * @param input
19       * @return string com o código
20       * @throws NoSuchAlgorithmException
21       */
22      public static String getSHA(String input) throws NoSuchAlgorithmException
23      {
24          // Static getInstance method is called with hashing SHA
25          MessageDigest md = MessageDigest.getInstance("SHA-256");
26
27          // digest() method called
28          // to calculate message digest of an input
29          // and return array of byte
30          byte[] hashCode = md.digest(input.getBytes(StandardCharsets.UTF_8));
31          return toHexString(hashCode);
32      }
33
34      /**
35       * passa o array de byte's para código hexadecimal, e depois para caracteres
36       * @param hash
37       * @return string de caracteres
38       */
39      private static String toHexString(byte[] hash)
40      {
41          // Convert byte array into signum representation
42          BigInteger number = new BigInteger(1, hash);
43
44          // Convert message digest into hex value
45          StringBuilder hexString = new StringBuilder(number.toString(16));
46
47          // Pad with leading zeros
48          while (hexString.length() < 32)
49          {
50              hexString.insert(0, '0');
51          }
52
53          return hexString.toString();
54      }
55  }

```

Anexo D - classe SHA

```

9 @Named("permissoesMB")
10 @ViewScoped
11 public class PermissoesManagedBean implements Serializable {
12
13     /**
14      *
15      */
16     private static final long serialVersionUID = 1L;
17
18     /**
19      * permissões na página [C, R, U, D]
20      *
21      * a área de consultas tem mais dois [C, R, U, D, I, P]
22      * - I Invalidar (cancelar) usado nas consultas
23      * - P Prescrição (pode passar/editar a prescrição)
24      * níveis
25      * A - Administrador
26      * S - Secretária (Administrativo)
27      * M - Médico
28      * P - Paciente
29      */
30     private boolean[] areaPacientes = {false, false, false, false};
31     private boolean[] areaMedicos = {false, false, false, false};
32     private boolean[] areaConsultas = {false, false, false, false, false, false};
33     private boolean[] areaUtilizadores = {false, false, false, false};
34
35     /**
36      * Construtor
37      * Guarda as permissões do utilizador definidas para o seu nível
38      */
39     public PermissoesManagedBean() {
40         SessaoManagedBean sessaoManagedBean = new SessaoManagedBean();
41
42         switch (sessaoManagedBean.getNivel()) {
43             case 'A':
44                 areaPacientes = new boolean[]{true, true, true, true};
45                 areaMedicos = new boolean[]{true, true, true, true};
46                 areaConsultas = new boolean[]{true, true, true, true, true, false};
47                 areaUtilizadores = new boolean[]{true, true, true, true};
48                 break;
49             case 'M':
50                 areaPacientes = new boolean[]{false, true, false, false};
51                 areaMedicos = new boolean[]{false, false, false, false};
52                 areaConsultas = new boolean[]{true, true, true, true, false, true};
53                 areaUtilizadores = new boolean[]{false, false, false, false};
54                 break;
55             case 'P':
56                 areaPacientes = new boolean[]{false, false, false, false};
57                 areaMedicos = new boolean[]{false, false, false, false};
58                 areaConsultas = new boolean[]{false, true, false, false, false, false};
59                 areaUtilizadores = new boolean[]{false, false, false, false};
60                 break;
61             case 'S':
62                 areaPacientes = new boolean[]{true, true, true, false};
63                 areaMedicos = new boolean[]{true, true, true, false};
64                 areaConsultas = new boolean[]{true, true, true, false, true, false};
65                 areaUtilizadores = new boolean[]{false, false, false, false};
66                 break;
67             default:
68                 break;
69         }
70     }
71
72     /**

```

Anexo E – parte do código da classe PermissoesManagedBean

```

1<html xmlns="http://www.w3.org/1999/xhtml"
2  xmlns:h="http://java.sun.com/jsf/html"
3  xmlns:ui="http://java.sun.com/jsf/facelets"
4  xmlns:b="http://bootsfaces.net/ui">
5
6<h:body>
7<b:navBar brand="#{agreementMB.marca}" brandHref="#" fluid="true" inverse="true">
8<b:navBarLinks>
9<b:navLink value="Home" href="/MyClinic"></b:navLink>
10
11<b:dropMenu rendered="#{permissoesMB.areaPacientes[1]}" value="Pacientes" iconAwesome="fas fa-user-friends">
12<b:navLink rendered="#{permissoesMB.areaPacientes[0] and sessaoMB.ligacaoBD}" value="Inserir" href="/MyClinic/pacientes/form"></b:navLink>
13<b:navLink value="Visualizar" href="/MyClinic/pacientes"></b:navLink>
14</b:dropMenu>
15
16<b:dropMenu rendered="#{permissoesMB.areaMedicos[1]}" value="Médicos" iconAwesome="fas fa-user-md">
17<b:navLink rendered="#{permissoesMB.areaMedicos[0] and sessaoMB.ligacaoBD}" value="Inserir" href="/MyClinic/medicos/form"></b:navLink>
18<b:navLink value="Visualizar" href="/MyClinic/medicos"></b:navLink>
19</b:dropMenu>
20
21<b:dropMenu rendered="#{permissoesMB.areaConsultas[1]}" value="Consultas" iconAwesome="fas fa-heartbeat">
22<b:navLink rendered="#{permissoesMB.areaConsultas[0] and sessaoMB.ligacaoBD}" value="Inserir" href="/MyClinic/consultas/form"></b:navLink>
23<b:navLink value="Visualizar" href="/MyClinic/consultas"></b:navLink>
24</b:dropMenu>
25
26<b:dropMenu rendered="#{permissoesMB.areaUtilizadores[0] and sessaoMB.ligacaoBD}" value="Utilizadores" iconAwesome="fas fa-user">
27<b:navLink value="Inserir" href="/MyClinic/utilizadores/form"></b:navLink>
28<b:navLink value="Visualizar" href="/MyClinic/utilizadores"></b:navLink>
29</b:dropMenu>
30</b:navBarLinks>
31
32<div style="float: right; display: inline-block;">
33<b:form class="form_fechar_sessao">
34
35<b:label severity="primary" text="Utilizador" />
36<b:label severity="secondary" text="#{sessaoMB.nomeUtilizador}" />
37<b:commandButton value="Sair" icon="fa-sign-out-alt" action="#{loginMB.logout()}"></b:commandButton>
38
39</b:form>
40</div>
41</b:navBar>
42</h:body>
43</html>

```

Anexo F – ficheiro menu.xhtml


```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
3 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
4 <html xmlns="http://www.w3.org/1999/xhtml"
5   xmlns:h="http://java.sun.com/jsf/html"
6   xmlns:ui="http://java.sun.com/jsf/facelets"
7   xmlns:p="http://primefaces.org/ui"
8   xmlns:f="http://java.sun.com/jsf/core"
9   xmlns:ib="http://bootsfaces.net/ui">
10   <h:body>
11     <ui:composition template="../../WEB-INF/templates/Layout.xhtml">
12       <ui:define name="content">
13         <h3 style="margin-top:0">Consultas</h3>
14         <b:form id="Form">
15           <p:dataTable var="consulta" value="#{consultaMB.consultaDAO.getLista()}" columnClasses="Label,value">
16             <p:column headerText="Paciente">
17               <h:outputText value="#{consulta.paciente.nome}" />
18             </p:column>
19             <p:column headerText="Medico">
20               <h:outputText value="#{consulta.medico.nome}" />
21             </p:column>
22             <p:column headerText="Especialidade">
23               <h:outputText value="#{consulta.medico.especialidade.descricao}" />
24             </p:column>
25             <p:column headerText="Data" style="width:120px">
26               <h:outputText value="#{consultaMB.dataFormatada(consulta.marcao)}" />
27             </p:column>
28             <p:column headerText="Valor">
29               <h:outputText value="#{consulta.valor}" />
30               <f:convertNumber type="currency" currencySymbol="€" maxIntegerDigits="5" maxFractionDigits="2"/>
31             </p:column>
32             <p:column headerText="Prescricao">
33               <h:outputText value="#{consulta.prescricao}" />
34             </p:column>
35             <p:column rendered="#{permissoesMB.areaConsultas[2]}" headerText="Ações" style="width:100px">
36               <f:param name="id" value="#{consulta.id}" />
37               <p:commandButton
38                 rendered="#{permissoesMB.areaConsultas[3] and not consulta.cancelada}" action="#{consultaMB.eliminar(consulta)}" icon="fa fa-trash">
39                 <p:confirm header="Confirmation" message="Confirma a eliminação?" icon="ui-icon-alert" />
40               </p:commandButton>
41               <p:confirmDialog global="true" showEffect="fade" hideEffect="fade">
42                 <p:commandButton value="Yes" type="button" styleClass="ui-confirmdialog-yes" icon="ui-icon-check" />
43                 <p:commandButton value="No" type="button" styleClass="ui-confirmdialog-no" icon="ui-icon-close" />
44               </p:confirmDialog>
45               <h:outputText value="Cancelada" rendered="#{consulta.cancelada}" />
46             </p:column>
47           </p:dataTable>
48         </b:form>
49       </ui:define>
50     </h:body>
51   </html>

```

Anexo G – ficheiro index.xhtml da consulta

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
3 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
4 <html xmlns="http://www.w3.org/1999/xhtml"
5   xmlns:h="http://java.sun.com/jsf/html"
6   xmlns:ui="http://java.sun.com/jsf/facelets"
7   xmlns:f="http://xmlns.jcp.org/jsf/core"
8   xmlns:ib="http://bootsfaces.net/ui">
9   <h:body onload="">
10     <ui:composition template="../../WEB-INF/templates/Layout.xhtml">
11       <ui:define name="content">
12         <b:container class="center"><h3 style="margin-top:0">Editar Consulta</h3></b:container>
13         <b:container style="margin:auto">
14           <b:container class="caixa_login" style="margin:auto;width:875px">
15             <b:form id="form" horizontal="true">
16               <b:selectOneMenu value="#{consultaMB.id_pacienteSelected}" required="true" requiredMessage="Paciente é obrigatório." label="Paciente:" colMd="10" labelColMd="2" select2="true">
17                 <f:selectItem itemLabel="(Seleccione)" itemValue="" />
18                 <f:selectItems value="#{consultaMB.pacienteDAO.getLista()}" var="c" itemValue="#{c.id}" itemLabel="#{c.nome}" />
19               </b:selectOneMenu>
20               <b:selectOneMenu value="#{consultaMB.id_medicoSelected}" required="true" requiredMessage="Médico é obrigatório." label="Médico:" colMd="10" labelColMd="2" select2="true">
21                 <f:selectItem itemLabel="(Seleccione)" itemValue="" />
22                 <f:selectItems value="#{consultaMB.medicoDAO.getLista()}" var="c" itemValue="#{c.id}" itemLabel="#{c.nome}" />
23               </b:selectOneMenu>
24               <b:dateTimePicker value="#{consultaMB.marcao}" required="true" requiredMessage="Marcação é obrigatória."
25                 allow-input-toggle="false" icon-position="Left" mode="" side-by-side="true"
26                 show-date="true" show-time="true" show-icon="true" use-current="true"
27                 format="YYYY-MM-DD HH:mm"
28                 placeholder="Marcação" label="Marcação:" colMd="10" labelColMd="2">
29                 <f:convertDateTime pattern="yyyy-MM-dd HH:mm"/>
30               </b:dateTimePicker>
31               <b:selectBooleanCheckbox rendered="#{permissoesMB.areaConsultas[4]}" value="#{consultaMB.consulta.cancelada}" caption="" label="Cancelar" colMd="10" labelColMd="2"/>
32               <b:inputText value="#{consultaMB.consulta.valor}" mask="9(1,5),99" placeholder="Preço" label="Preço:" colMd="10" labelColMd="2">
33                 <f:convertNumber maxIntegerDigits="5" maxFractionDigits="2" groupingUsed="false"/>
34               </b:inputText>
35               <b:inputTextArea rendered="#{permissoesMB.areaConsultas[5]}" value="#{consultaMB.consulta.prescricao}" placeholder="Prescrição" label="Prescrição:" colMd="10" labelColMd="2"/>
36             </b:form>
37             <b:messages id="msgs" style="text-align:left"/>
38             <b:formGroup style="text-align:right;margin: 0">
39               <b:commandButton label="" value="Atualizar" id="btnForm" update="@form" process="@form" action="#{consultaMB.atualizar()}" icon="ui-icon-star" style="width:92px;"></b:commandButton>
40             </b:formGroup>
41           </b:container>
42         </b:container>
43       </ui:define>
44     </h:body>
45   </html>

```

Anexo H – ficheiro editar.xhtml da consulta

MyClinic

Home

Pacientes

Médicos

Consultas

Utilizadores

Utilizador

admin

Sair

Novo Utilizador

Utilizador: *

Palavra-Passe: *

Confirmação: *

Nível: *

(Selecione)

(Selecione)






Paciente

Médico

Administrativo

Administrador

© 2019 MyClinic All rights reserved

Siga-nos:     

Anexo I – página de inserir novo utilizador