

## **Projeto Final – M2**

### **Sistema: GPS for Agents**

Curso: ANÁLISE E DESENVOLVIMENTO DE SISTEMAS – 4º Semestre

Disciplina: ANÁLISE E IMPLEMENTAÇÃO DE SOFTWARE - 4°C

Professor: Adilson

Alunos:

- Filipe Lourenço – RGM: 11241103720
- Caio Filipi dos Santos da Silva – RGM: 11241103315

### **1. Introdução**

Aqui descrevemos o desenvolvimento do sistema GPS for Agents, uma aplicação projetada para apoiar o trabalho dos agentes comunitários de saúde, utilizando recursos de geolocalização, registro territorial e consulta de informações operacionais.

O projeto conta com programação orientada a objetos, integração com banco de dados, arquitetura cliente-servidor e utilização de APIs externas. O objetivo é demonstrar não apenas a implementação funcional do sistema, mas também a capacidade de planejar, modelar e documentar soluções de software alinhadas às dores reais.

### **2. Descrição do Sistema – GPS for Agents**

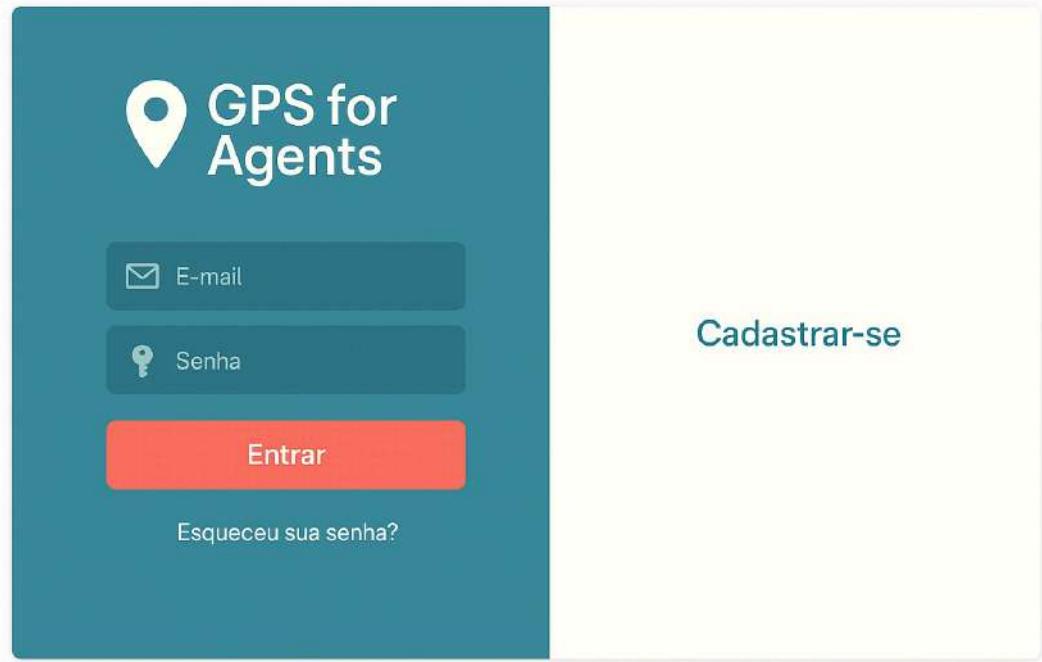
O sistema GPS for Agents permite ao usuário autenticar-se, visualizar um mapa interativo, registrar marcadores personalizados, inserir anotações associadas ao perfil e interagir com a interface de forma intuitiva. A solução foi pensada para facilitar o cotidiano dos agentes de saúde, organizando rotas, pontos de visita e registros informativos.

### **3. Tecnologias Utilizadas**

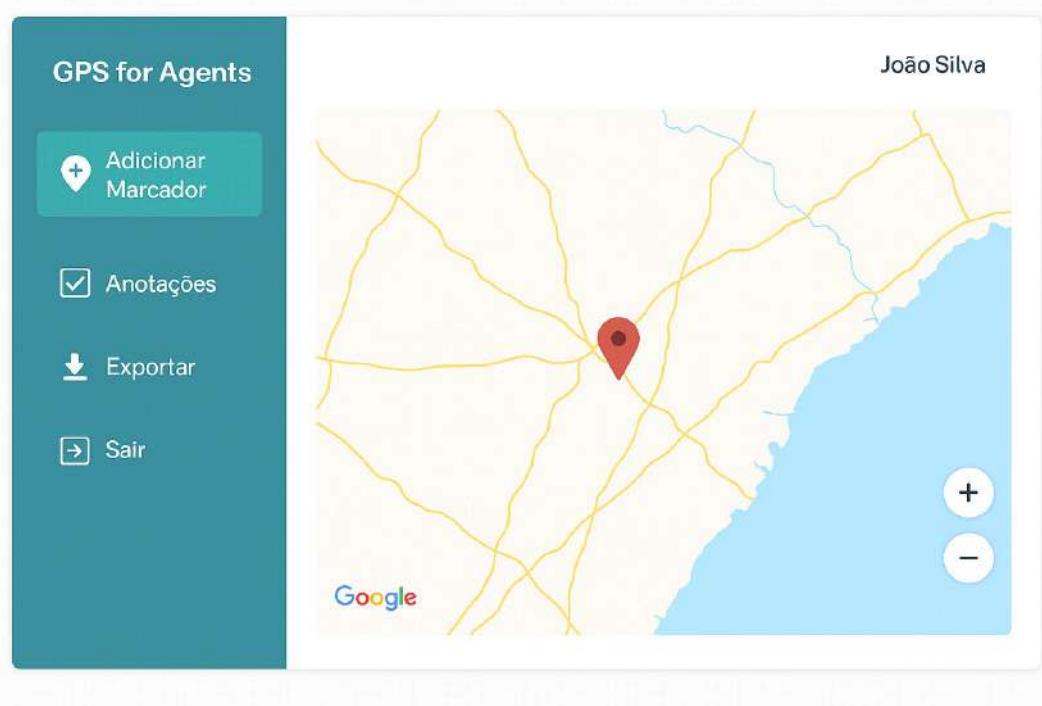
- Java (JSP/Servlets)
- HTML, CSS e JavaScript
- Bootstrap 5
- Banco MySQL
- Servidor Apache Tomcat
- API Google Maps
- NetBeans IDE
- GitHub para versionamento

### **4. Desenho / Design da Solução**

A seguir, apresenta-se o diagrama prototípico de arquitetura .



Cadastrar-se



## 5. Banco de Dados

O banco de dados foi estruturado de forma eficiente, com três entidades principais: usuários, marcadores e anotações. A modelagem visa garantir organização e consistência durante o armazenamento das informações.

### Script SQL:

The screenshot displays two instances of the phpMyAdmin interface. The top instance shows the database structure for the 'gofrageiro' schema, featuring tables like 'clients', 'images', 'marcadores', and 'users'. The bottom instance shows the results of a query executed in the 'Consultas' tab, displaying a table of file attachments with columns: ID, ID\_usuario, nome, tipo, dados, and ctsdt\_um. The results show four rows of attachments, each with a 'Visualizar' link and a 'Remover' button.

ID	ID_usuario	nome	tipo	dados	ctsdt_um
5	3	0_Logo_GPS.png	image/png	30,0B - 66,3 KB	2023-11-26 16:27:23
6	3	0_desk_sistema.png	image/png	30,0B - 291,6 KB	2023-11-26 16:29:00
7	3	0_Logo_gente(1).png	image/png	30,0B - 76,9 KB	2023-11-26 16:45:03
8	3	0_rega_usuário(1)(1).png	image/png	30,0B - 154,0 KB	2023-11-27 19:31:15

**Table: clients**

	<b>id</b>	<b>nome</b>	<b>email</b>	<b>cep</b>	<b>senha</b>	<b>foto</b>	<b>foto_content_type</b>
1	1	Flávia Cesar	flavia.cesar55@gmail.com	08900100-123	EB1DB-67-84E5	image/png	
2	2	Teste da Tábu	teste@gmail.com	0105020-000	VULZ	AVAI	
3	3	Cleopatra	Compravida@bancos.com.br	08890010-000	Camporosa11	NULZ	

**Table: residencies**

	<b>id</b>	<b>M_residencia</b>	<b>Id</b>	<b>Ing</b>	<b>Reta</b>	<b>date_cadastro</b>	<b>Obj</b>
1	1	Editor	42	Copiar	Remover	2023-11-20 11:23:41	Casa
2	2	Editor	43	Copiar	Remover	2023-11-20 14:14:54	Casa
3	3	Editor	44	Copiar	Remover	2023-11-20 14:46:11	Casa
4	4	Editor	45	Copiar	Remover	2023-11-20 20:03:52	par
5	5	Editor	46	Copiar	Remover	2023-11-20 21:01:47	Casa

## 6. Evidências do Sistema em Execução

GPS for Agents - Landing Page

localhost:8080/AgentesGPS/

Cadastrar | Logout | ☽

## Simplifique a rotina dos Agentes de Saúde

Organize visitas, mapeie regiões, acompanhe famílias e gerencie campanhas com uma plataforma moderna criada especialmente para Agentes Comunitários.

### Funcionalidades Principais

- Mapeamento Urbano Inteligente
- Cadastro de Famílias
- Planejamento de Visitas
- Relatórios Rápidos
- Acompanhamento de Morbidades

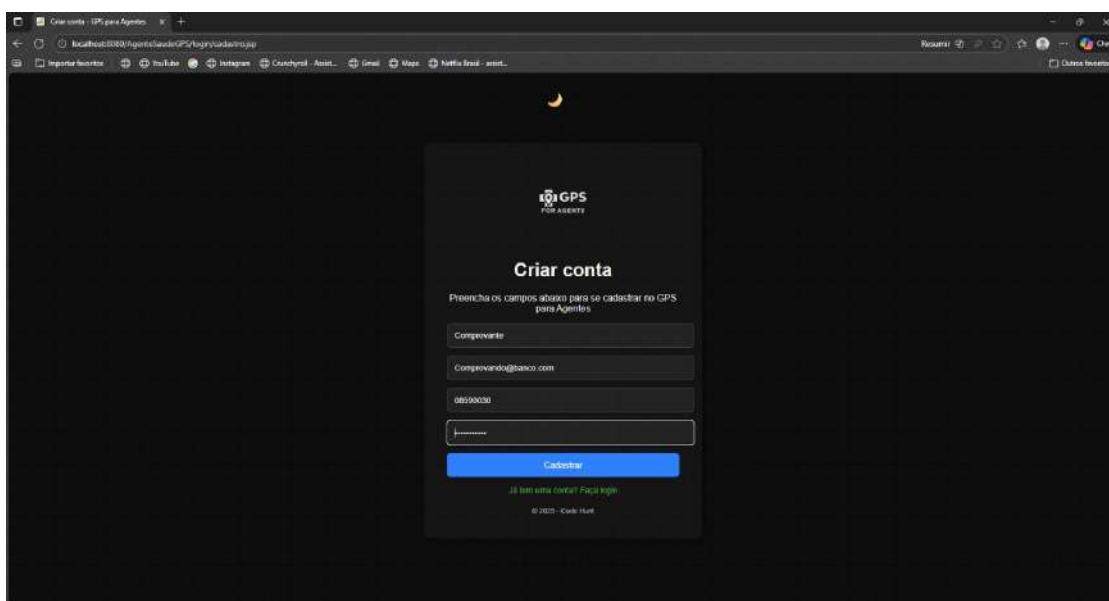
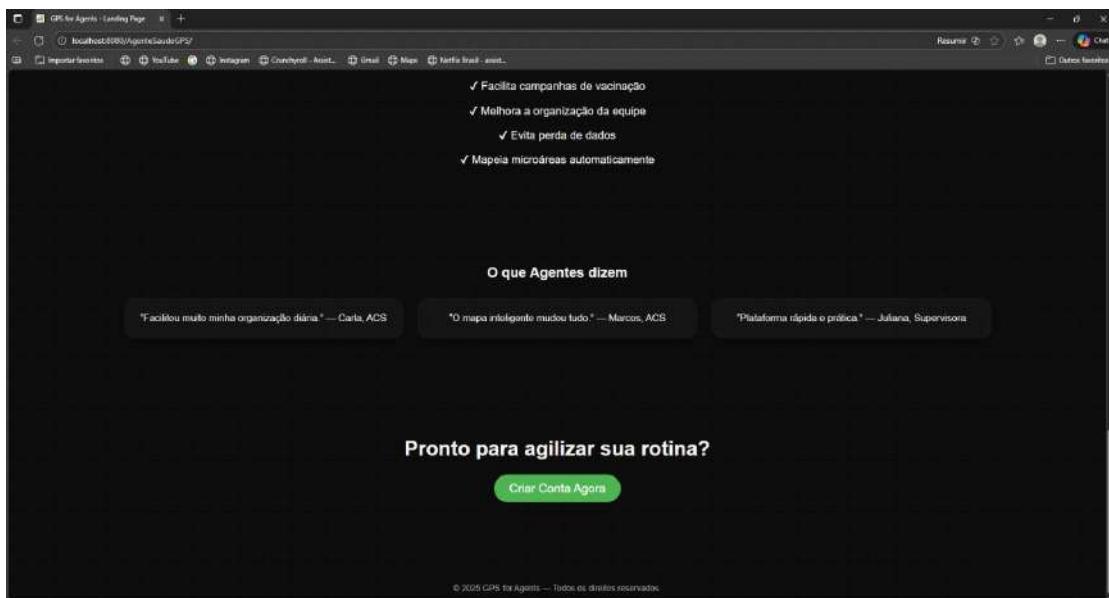
GPS for Agents - Landing Page

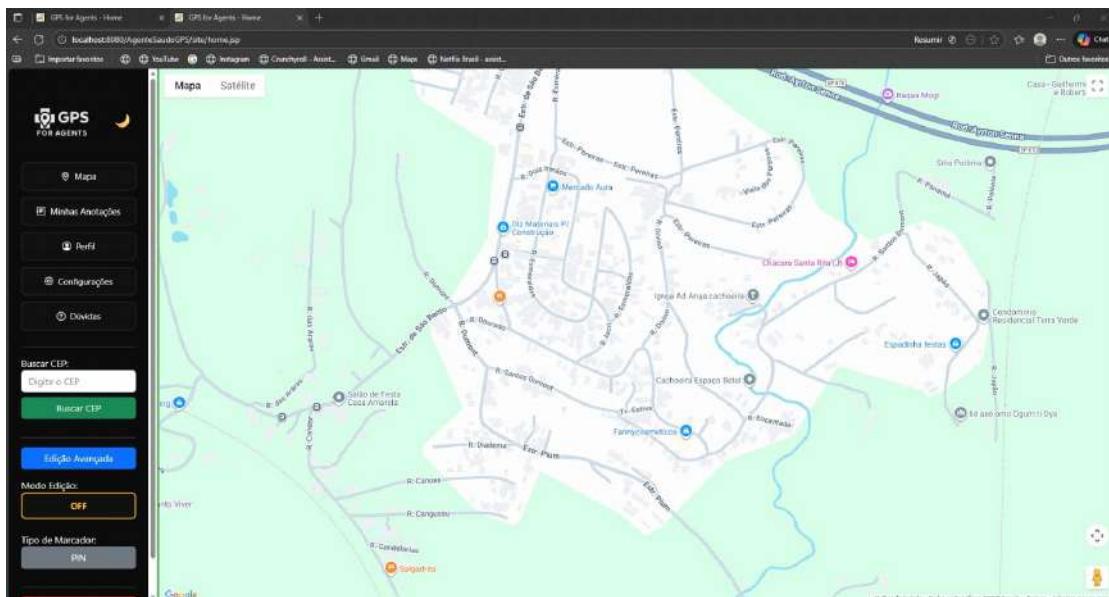
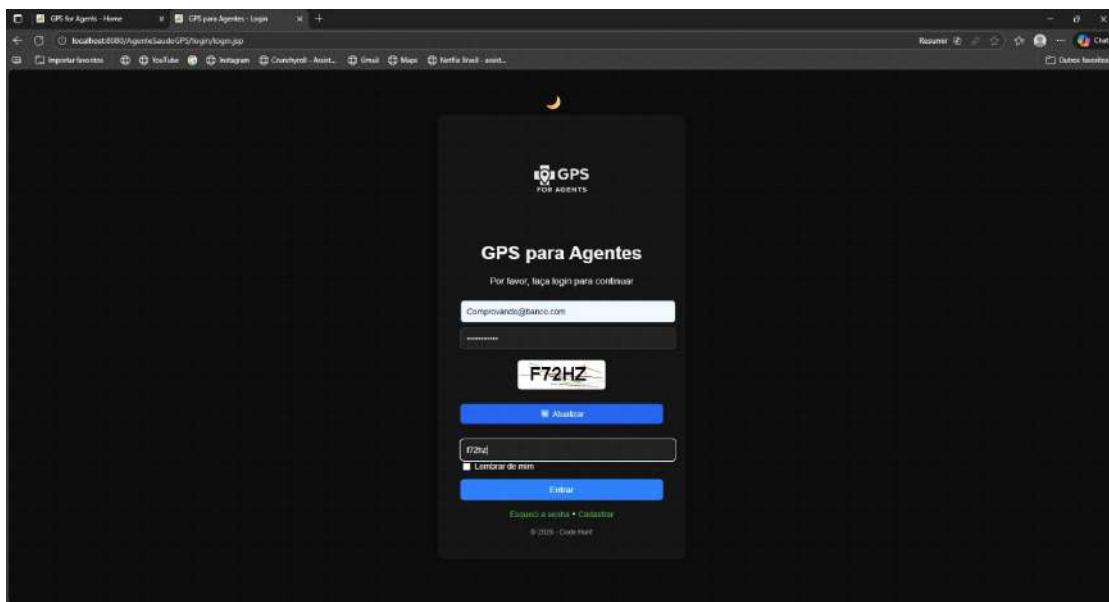
localhost:8080/AgentesGPS/

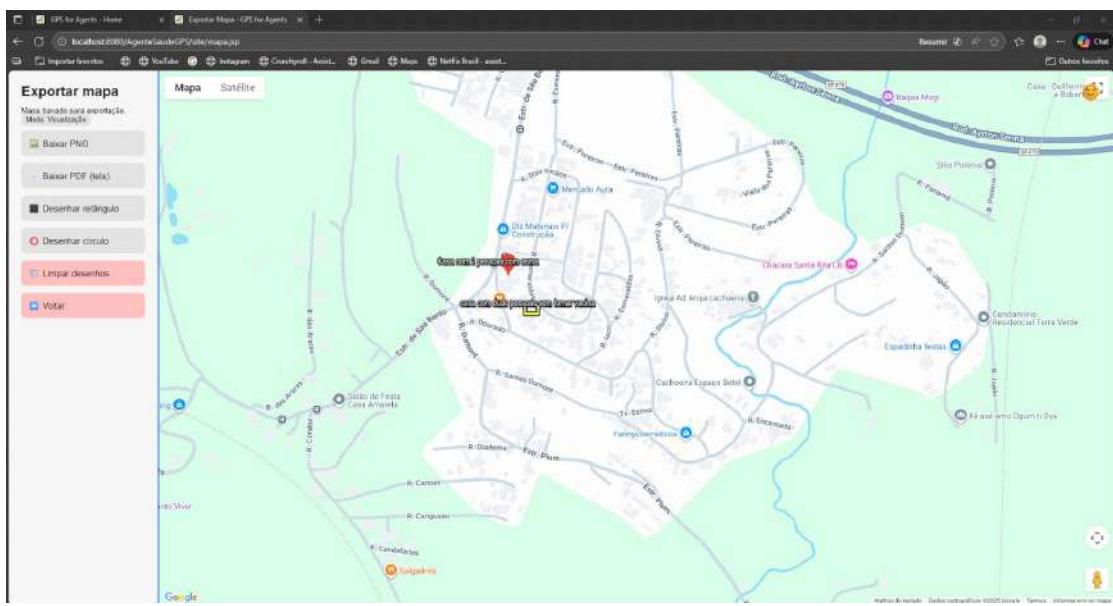
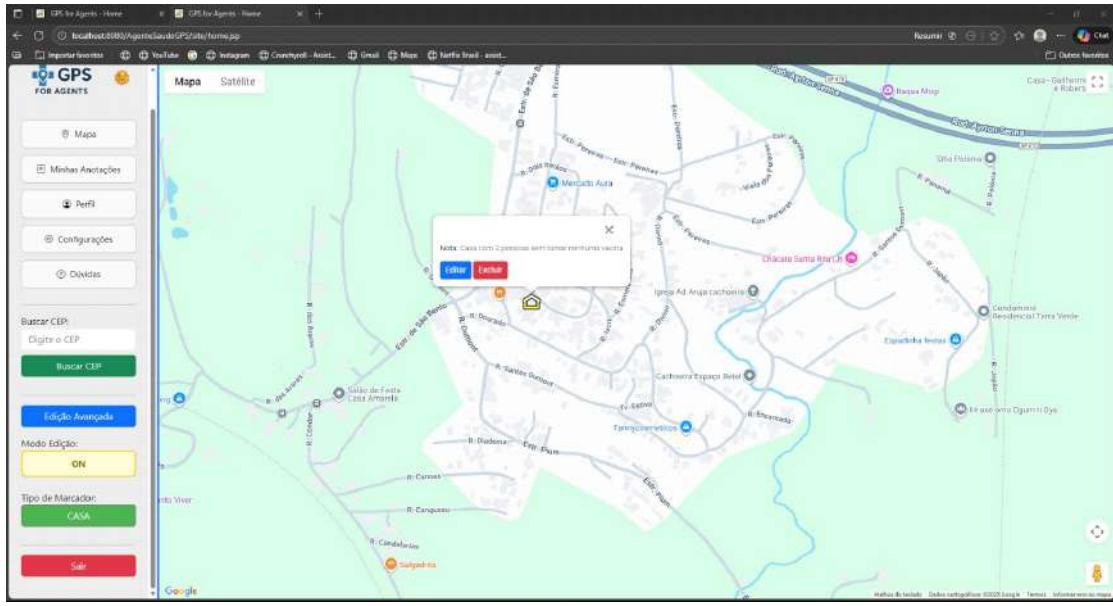
### Veja como o sistema funciona

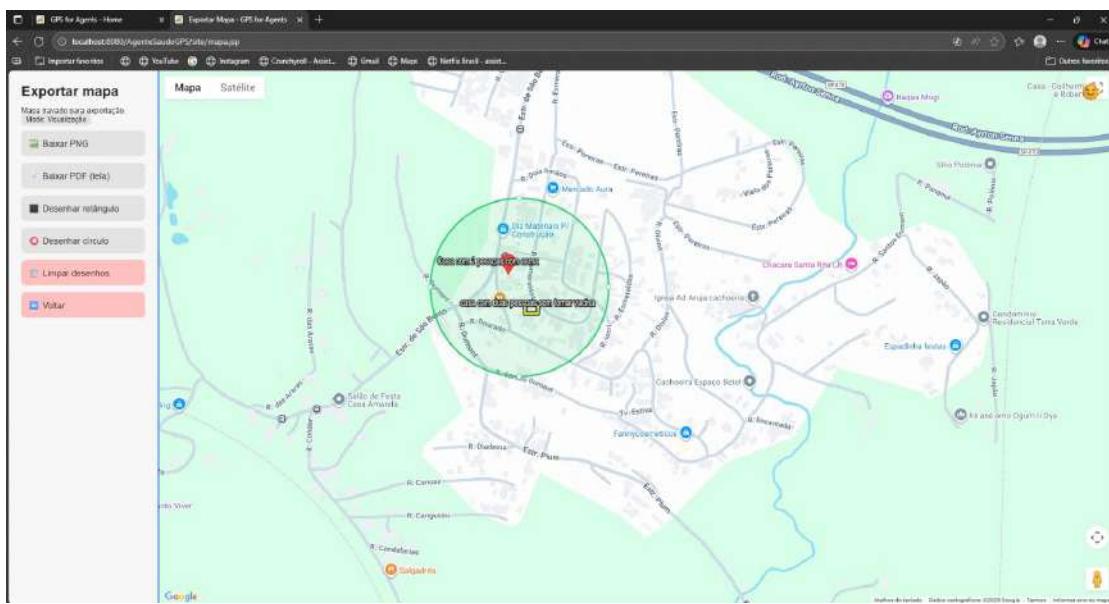
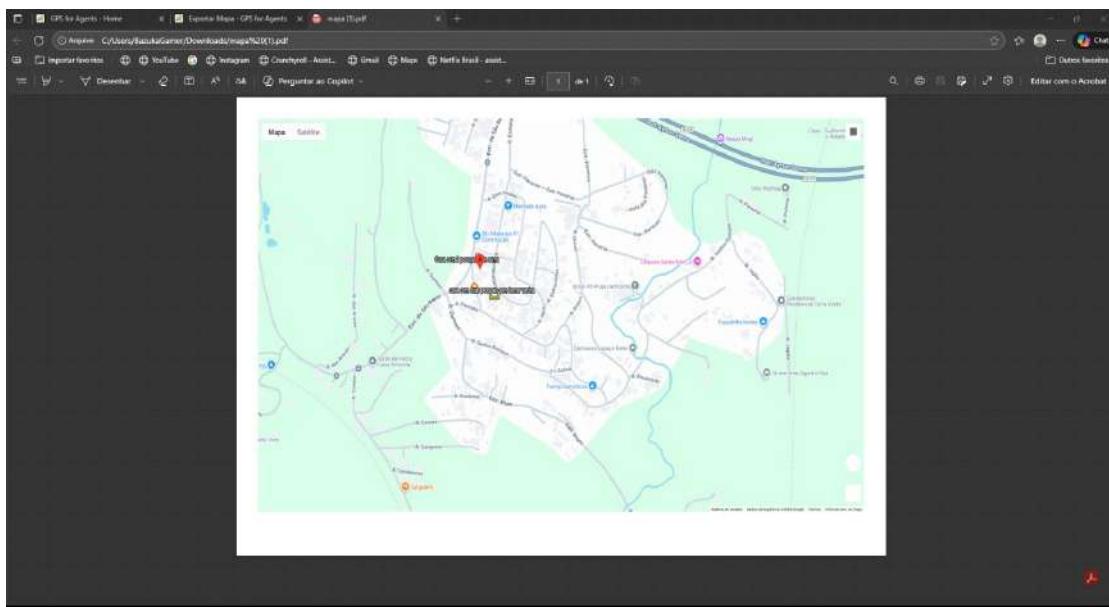
Exportar mapa

- Rota intercâmbio residência
- Rota residência
- Salvar PNG
- Salvar PDF (pdf)
- Desenhar retângulo
- Desenhar círculo
- Limpar desenho
- Voltar









GPS for Agents - Home

localhost:8080/Agentes/areaGPS/info/anotações.jsp

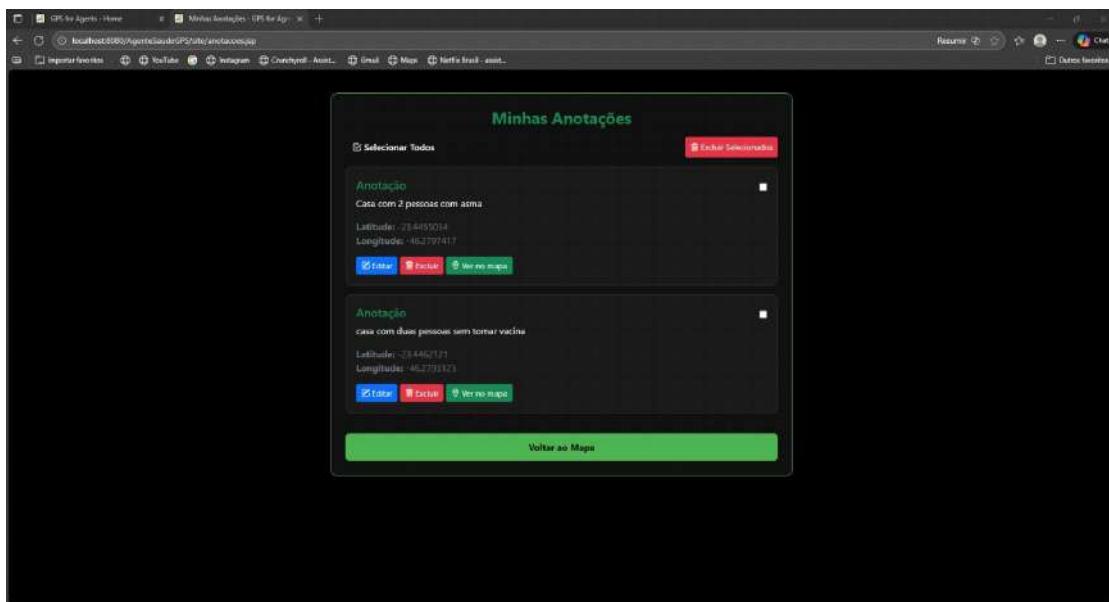
Minhas Anotações

Selecionar Todos Excluir Selecionadas

**Anotação**  
Casa com 2 pessoas com sono  
Latitude: -23.4485034  
Longitude: -46.2797417  
Fazer Editar Ver no mapa

**Anotação**  
casa com duas pessoas sem tomar vacina  
Latitude: -23.4485034  
Longitude: -46.2797417  
Fazer Editar Ver no mapa

[Voltar ao Mapa](#)



GPS for Agents - Home

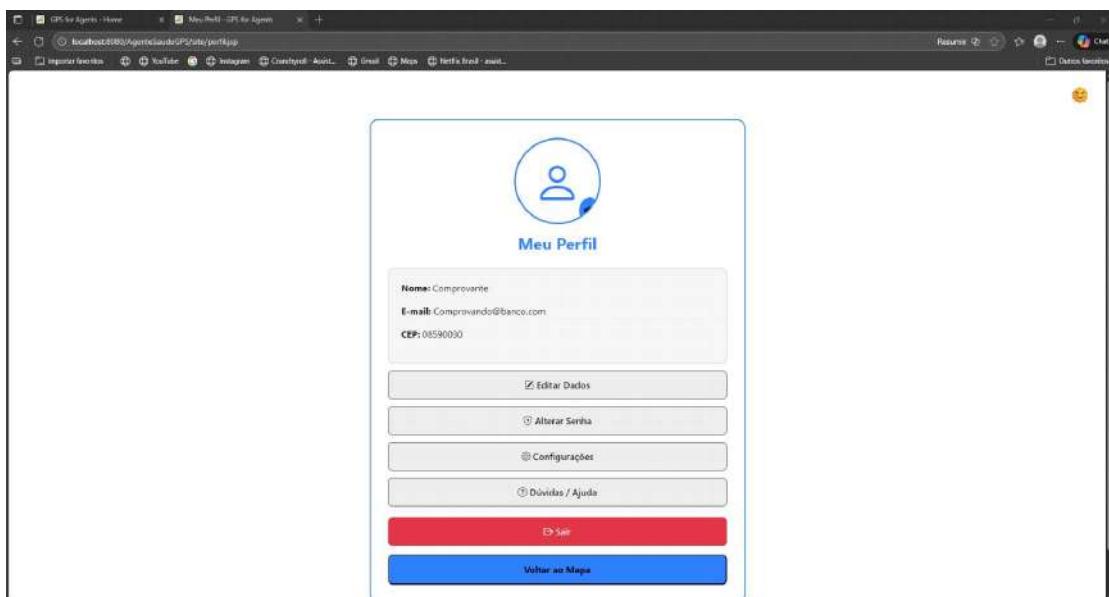
localhost:8080/Agentes/areaGPS/controle/perfil.jsp

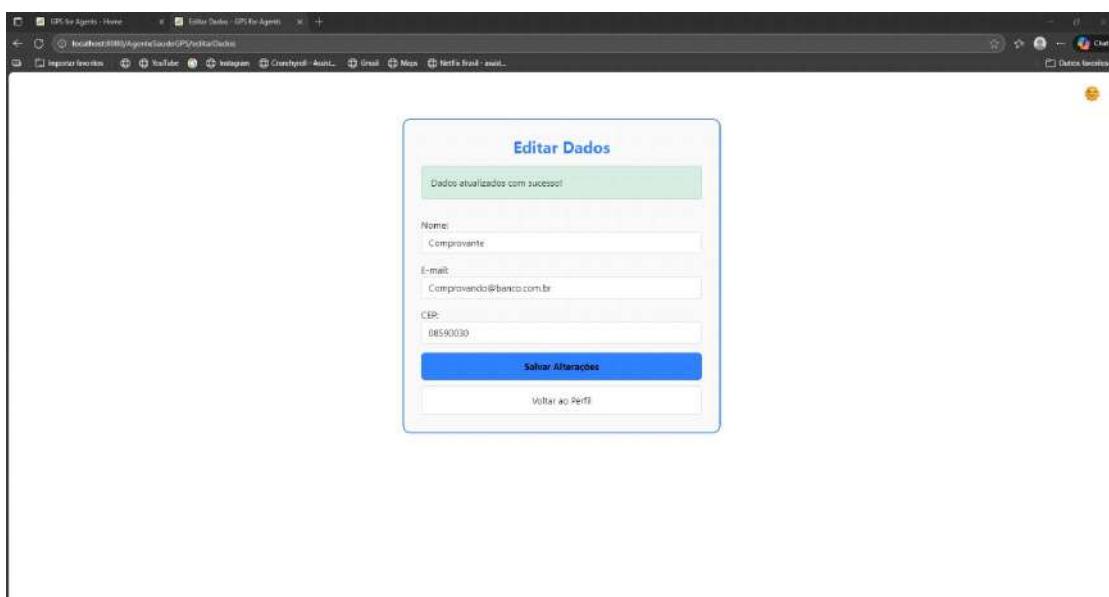
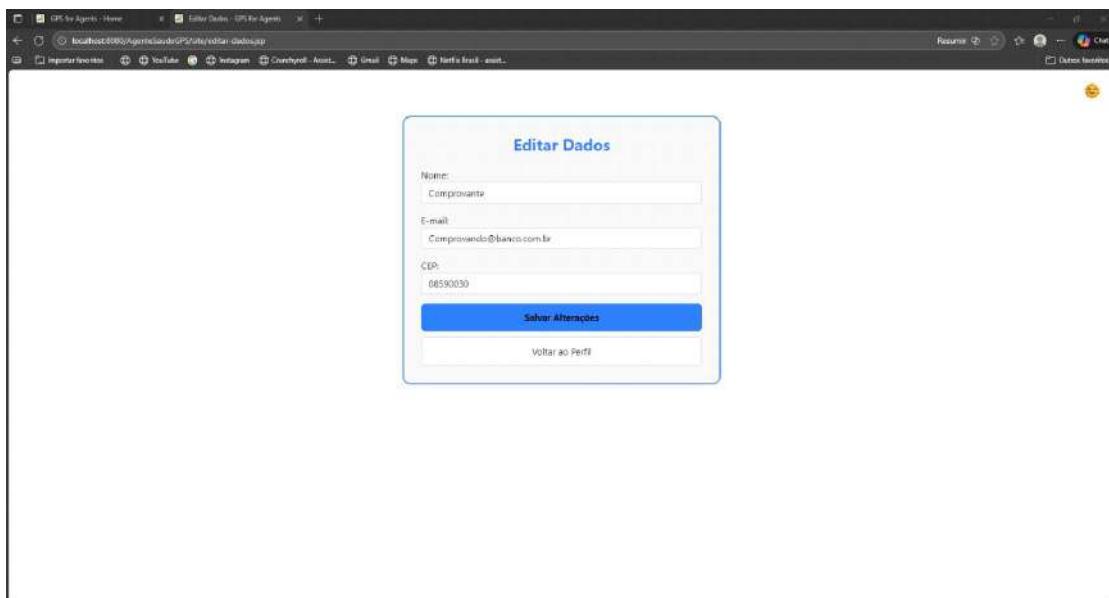
Meu Perfil

  
Nome: Compravante  
E-mail: Compravante@banco.com  
CEP: 08590090

Editar Dados  
 Alterar Senha  
 Configurações  
 Dúvidas / Ajuda

Sair Voltar ao Mapa





CPS for Agents - Home

localhost:8080/AgentCloudsGPS/WEB/alterar-senha.jsp

Importar favoritos

Outras favoritas

Resumir

Chat

Outras telas

Smiley

### Alterar Senha

Senha atual:

Nova senha:

Confirmar nova senha:

**Salvar nova senha**

[Voltar ao Perfil](#)

CPS for Agents - Home

localhost:8080/AgentCloudsGPS/alterar-senha

Importar favoritos

Outras favoritas

Resumir

Chat

Outras telas

Smiley

### Alterar Senha

Senha alterada com sucesso!

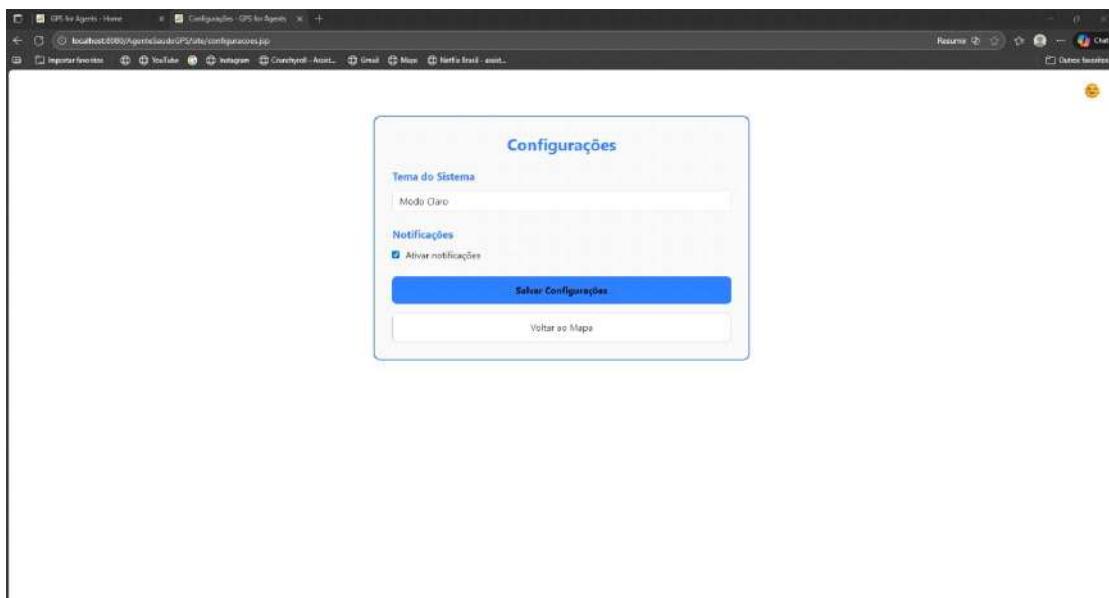
Senha atual:

Nova senha:

Confirmar nova senha:

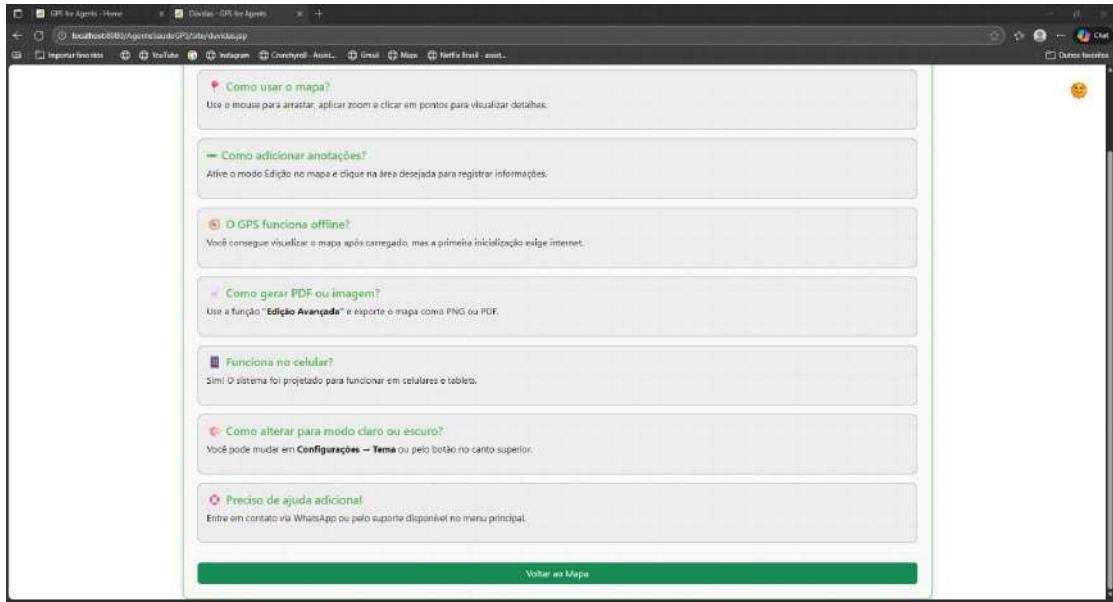
**Salvar nova senha**

[Voltar ao Perfil](#)



A screenshot of a web browser showing the 'Central de Dúvidas' (Help Center) page of the GPS for Agents application. The page has a green header with the title 'Central de Dúvidas' and a sub-instruction: 'Aqui você encontra respostas rápidas sobre o uso do GPS for Agents.' Below the header is a list of frequently asked questions (FAQs) in boxes:

- **Como usar o mapa?**  
Use o mouse para anotar, aplicar zoom e clicar em pontos para visualizar detalhes.
- **Como adicionar anotações?**  
Ative o modo Edição no mapa e clique na área desejada para registrar informações.
- **O GPS funciona offline?**  
Você consegue visualizar o mapa após carregado, mas a primeira inicialização exige internet.
- **Como gerar PDF ou imagem?**  
Use a função "**Edição Avançada**" e exporte o mapa como PNG ou PDF.
- **Funciona no celular?**  
Sim! O sistema foi projetado para funcionar em celulares e tablets.
- **Como alterar para modo claro ou escuro?**  
Você pode mudar em **Configurações - Tema** ou pelo botão no canto superior.
- **Precise de ajuda adicional**



index.jsp

The screenshot shows the Eclipse IDE interface with the Java code for the `LandingPageController.java` file. The code defines a controller for the landing page, handling requests for the URL `/agentes`. It uses Thymeleaf to render the `landing.html` template, which displays a list of agents with their names and descriptions. The code also includes logic to handle file uploads for agent images.

```
Document - landing.html
Created on: 27/12/2012
Author: jorge
Private w/ Class
Description:
    Landing Page principal do GFS para agentes.
    HTML
    - Autenticável ou sistema
    - Descrição
    - Fazia download
    - Descrição
    - Listas de agentes e contatos
    - Integração com o tema global (rotinas + localizações)
<html>
<head>
    <meta charset="UTF-8" />
    <title>GFS - Agentes - Landing Page</title>
</head>
<body>
    <div>Landing Page principal do GFS para agentes.</div>
    <table border="1">
        <thead>
            <tr>
                <th>Nome do Agente</th>
                <th>Descrição do Agente</th>
                <th>Arquivo de Imagem</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>Agentes001</td>
                <td>Autenticável ou sistema</td>
                <td><a href="#">Download</a></td>
            </tr>
            <tr>
                <td>Agentes002</td>
                <td>Descrição</td>
                <td><a href="#">Download</a></td>
            </tr>
            <tr>
                <td>Agentes003</td>
                <td>Fazia download</td>
                <td><a href="#">Download</a></td>
            </tr>
            <tr>
                <td>Agentes004</td>
                <td>Descrição</td>
                <td><a href="#">Download</a></td>
            </tr>
            <tr>
                <td>Agentes005</td>
                <td>Listas de agentes e contatos</td>
                <td><a href="#">Download</a></td>
            </tr>
            <tr>
                <td>Agentes006</td>
                <td>Integração com o tema global (rotinas + localizações)</td>
                <td><a href="#">Download</a></td>
            </tr>
        </tbody>
    </table>
</body>
</html>
```





The screenshot shows the AgentStudio IDE interface with the following details:

- Project Tree:** Shows packages like `AgentServiceGP`, `WebPages`, `Controller`, `Model`, and `Util`. Under `Controller`, there are several controller classes: `CalculatorController.java`, `CaptchaController.java`, `ConfigurationController.java`, `CustomerController.java`, `EmailController.java`, `FontController.java`, `ImageController.java`, `LanguageController.java`, `ManadoController.java`, `RecipientController.java`, `VirtualPhoneController.java`, and `WarehouseController.java`.
- Code Editor:** The main editor window displays Java code for the `CalculatorController.java` class. The code includes imports for `java.util.List`, `java.util.Map`, `java.util.ResourceBundle`, and `java.util.concurrent.ConcurrentHashMap`. It defines a class `CalculatorController` with methods `getCalculator()`, `getCalculatorList()`, and `getCalculatorCount()`. The `getCalculator()` method returns a `Map<String, Object>` containing a `ResourceBundle` and a `ConcurrentHashMap` for calculations.

The screenshot shows the Apache NetBeans IDE interface with the following details:

- Project Tree:** The "ApacheGaucho" project is selected. It contains several Java files (e.g., Application.java, Controller.java, ImageController.java, Main.java) and a CSS file named "style.css".
- Code Editor:** The "style.css" file is open in the main editor area. The code defines a class ".header" with various styling rules like background-color, border-radius, and box-shadow. It also includes a ".header .nav-item" rule and a ".header .nav-item .nav-link" rule.
- Toolbars:** Standard NetBeans toolbars for File, Edit, View, Navigator, Source, Refactor, Run, Debug, Profile, Tools, and Window are visible at the top.
- Status Bar:** The status bar at the bottom shows "ApacheGaucho - Apache NetBeans 12.3" and "File View Help".



The screenshot shows the NetBeans IDE interface with the following details:

- Project:** AgentesSaudeOPS
- File Structure:** Shows the project structure with modules like AgentesSaudeOPS, WebPages, controllers, models, and DAOs.
- Code Editor:** Displays the content of the file `index.jsp`. The code includes JSP tags, CSS classes, and Java beans. It features sections for "PAINEL DE USUÁRIOS", "PAINEL DE FUNCIONÁRIOS", "PAINEL DE SISTEMA", and "NOTÍCIAS". A comment at the bottom asks "O que os Agentes dizem?".
- Toolbars:** Standard NetBeans toolbars for file operations, code navigation, and search.
- Status Bar:** Shows the current file path as "index.jsp - Apache NetBeans (15)" and the line number "151".

ConectaDB

```
Project: AgentSauvage
File Edit View Navigate Source Refactor Run Debug Profile Tools Tasks WebHelp Help
AgentSauvage - Apache NetBeans 12.5
Search Ctrl+F

Source History Diff Commit Log
ConexaoDB.java - calendar.jsp - index.jsp

Source
1 package entity;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.SQLException;
6
7 /**
8 * Classe: ConexaoDB
9 * Autor : Felipe C. Caldeira
10 * Descrição:
11 * Permite conexão com o MySQL utilizando o driver JDBC anterior
12 * que importa a URL e a classe de configuração
13 */
14
15 public class ConexaoDB {
16
17     public static Connection conectar() throws ClassNotFoundException {
18
19         Connection conn = null;
20
21         try {
22             // Informações do MySQL
23             Class.forName("com.mysql.cj.jdbc.Driver");
24
25             // URL com parâmetros para evitar problemas de charset e timeout
26             String url =
27                 "jdbc:mysql://localhost:3306/AgenteSauvage?"
28                 + "useUnicode=true"
29                 + "characterEncoding=UTF-8"
30                 + "improvedSQLiteWAL=true"
31                 + "useSSL=false"
32                 + "allowPublicKeyRetrieval=true"
33                 + "trustServerCertificate=true";
34
35             // Início da conexão com o banco
36             conn = DriverManager.getConnection(url, "root", "1234");
37
38         } catch (SQLException ex) {
39             System.out.println("Erro de conexão com o banco: " + ex.getMessage());
40         }
41
42         return conn;
43     }
44
45 }

config.ConexaoDB > connector > by
```

Cliente.java

Client.java

```
package models;

public class Client {
    private int id;
    private String name;
    private String email;
    private String phone;
    private String gender;

    // CONSTRUCTORS
    public Client(int p_id) { this.id = p_id; }
    public Client(String p_name) { this.name = p_name; }
    public Client(String p_email) { this.email = p_email; }
    public Client(String p_phone) { this.phone = p_phone; }
    public Client(String p_gender) { this.gender = p_gender; }

    // GETTERS
    public int getId() { return this.id; }
    public String getName() { return this.name; }
    public String getEmail() { return this.email; }
    public String getPhone() { return this.phone; }
    public String getGender() { return this.gender; }
}
```

## Imagen.java

Imagen.java

```
package models;

public class Imagen {
    private int id;
    private int referencia;
    private String nome;
    private String tipo;
    private byte[] dados;

    // CONSTRUCTORS
    public Imagen() {
        this.id = 0;
    }
    public Imagen(int id) {
        this.id = id;
    }

    // GETTERS AND SETTERS
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }

    public int getReferencia() {
        return referencia;
    }
    public void setReferencia(int referencia) {
        this.referencia = referencia;
    }

    public String getNome() {
        return nome;
    }
    public void setNome(String nome) {
        this.nome = nome;
    }

    public String getTipo() {
        return tipo;
    }
    public void setTipo(String tipo) {
        this.tipo = tipo;
    }

    public byte[] getDados() {
        return dados;
    }
    public void setDados(byte[] dados) {
        this.dados = dados;
    }
}
```

Marcador.java

The screenshot shows the Eclipse IDE interface with the following details:

- File Bar:** File, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Project Explorer:** Shows a Java project named "paciente-api4.5.1-pseudos".
- Java Editor:** Displays the source code for the "Mentorador" class. The code includes imports for `java.util.List`, `java.util.ArrayList`, and `java.util.LinkedList`. The class has a private attribute `listMentor` and a constructor that initializes it to an empty list. It contains methods `getMentor()` and `actionMentor(Usuario usuario)`.
- Search Bar:** Search Ctrl+Shift+F.

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The left sidebar displays the project structure under "Projects". It includes modules like "jewelry-store-api-0.0.1", "jewelry-store", "src", "config", "model", "modelMapper", and "util".
- Code Editor:** The main window shows the Java code for a controller class named "Controller.java". The code includes methods for setting and getting a double value, and a method for setting a note.
- Toolbars and Status Bar:** The top bar has standard file menu items (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Windows, Help). The status bar at the bottom right shows "11:31 105. Windows [ORL]".

```
public void setMilestoneId(int id) {
    this.milestoneId = id;
}

// To send back basic milestone
public int getMilestoneId() {
    return milestoneId;
}

public void setMilestoneName(String name) {
    this.milestoneName = name;
}

public String getName() {
    return name;
}

public void setLastEditDate(Date lastEdit) {
    this.lastEdit = lastEdit;
}

// Lengths
public double getLength() {
    return length;
}

public void setLength(double length) {
    this.length = length;
}

// Marmada milestones or association
public String getMarmada() {
    return marmada;
}

public void setMarmada(String marmada) {
    this.marmada = marmada;
}

// Date format is seconds (int "sec", "secs")
public String getTstamp() {
    return tstamp;
}

public void setTstamp(String tstamp) {
    this.tstamp = tstamp;
}
```

## ClientesDAO.java



The screenshot shows the NetBeans IDE interface with the following details:

- Project Tree:** Shows a project named "pacientes-api-4.0.1-producer" containing packages like "model", "rest", "entity", "util", "dao", and "controller".
- Code Editor:** Displays the `ClientController.java` file. The code implements the `IClientController` interface and extends `BaseController<IClient>`. It includes methods for creating, updating, deleting, and finding clients, as well as methods for generating QR codes and sending emails.
- Toolbars:** Standard NetBeans toolbars for file operations, search, and navigation.
- Status Bar:** Shows the path "src/main/java/com/raul/raul/pacientes/api/4.0.1/controller" and the status "Synced (100%) - Apache NetBeans (DE-7)".

```
private void ATUALIZARSENHA(CLIENTE cli) throws ClienteNotFoundException {
    String sql = "UPDATE clientes SET senha = ?, email = ?, msg = ? WHERE id = ?";
    try (Connection conn = ConexaoBD.getConnection();
        PreparedStatement stmt = conn.prepareStatement(sql)) {
        stmt.setString(1, cli.getSenha());
        stmt.setString(2, cli.getEmail());
        stmt.setString(3, cli.getMsg());
        stmt.setInt(4, cli.getId());
        stmt.executeUpdate();
    } catch (SQLException e) {
        System.out.println("Erro ao atualizar senha: " + e.getMessage());
    }
}

public void alterarSenha(CLIENTE cliente) throws ClienteNotFoundException {
    String sql = "UPDATE clientes SET senha = ? WHERE id = ?";
    try (Connection conn = ConexaoBD.getConnection();
        PreparedStatement stmt = conn.prepareStatement(sql)) {
        stmt.setString(1, cliente.getSenha());
        stmt.setInt(2, cliente.getId());
        stmt.executeUpdate();
    } catch (SQLException e) {
        System.out.println("Erro ao alterar senha: " + e.getMessage());
    }
}
```

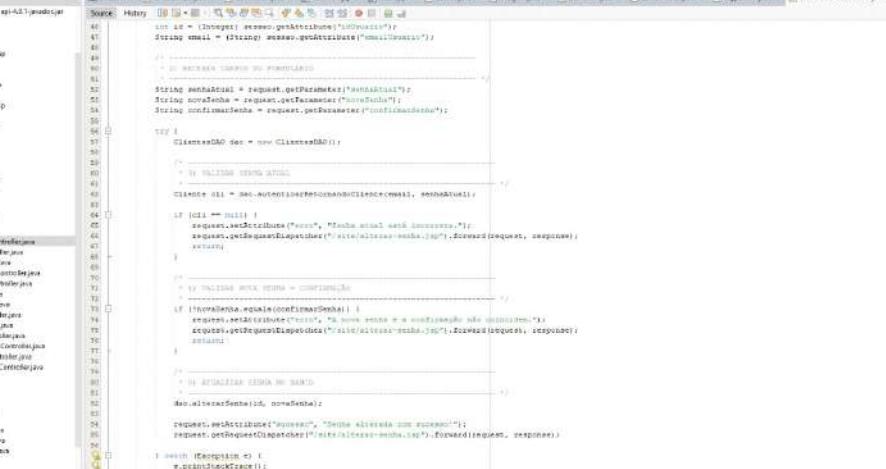
## CryptoUtil

```
private static final String SECRET_KEY = "1234567890ABCDE"; // 16 bytes
private static final String INIT_VECTOR = "P@ssw0rd12345678"; // 16 bytes

public static byte[] encrypt(byte[] data) throws Exception {
    IvParameterSpec iv = new IvParameterSpec(INIT_VECTOR.getBytes("UTF-8"));
    SecretKeySpec key = new SecretKeySpec(SECRET_KEY.getBytes("UTF-8"), "AES");
    Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
    cipher.init(Cipher.ENCRYPT_MODE, key, iv);
    return cipher.doFinal(data);
}

public static byte[] decrypt(byte[] encrypted) throws Exception {
    IvParameterSpec iv = new IvParameterSpec(INIT_VECTOR.getBytes("UTF-8"));
    SecretKeySpec key = new SecretKeySpec(SECRET_KEY.getBytes("UTF-8"), "AES");
    Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
    cipher.init(Cipher.DECRYPT_MODE, key, iv);
    return cipher.doFinal(encrypted);
}
```

## AlterarSenhaController.java



The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (left):** Shows the project structure with packages like `com.aplicativo.api.v1.productos`, `com.aplicativo.controllers`, `com.aplicativo.services`, and `com.aplicativo.utils`. Under `com.aplicativo.controllers`, files like `AlimentoController.java`, `CategoriaController.java`, `ConfiguracionController.java`, `EdicionController.java`, `FotoController.java`, `ImagenController.java`, `LeyController.java`, `MarcaController.java`, `RequerimientoController.java`, `UploadingController.java`, and `UploadingImageController.java` are listed.
- Code Editor (center):** Displays Java code for the `AlimentoController.java` file. The code handles requests for adding, updating, and deleting alimentos, interacting with the `AlimentoService` and `AlimentoDAO` via `AlimentoMapper`.
- Toolbars and Menus (top):** Standard Eclipse menus like File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help are visible. The toolbar includes icons for New, Open, Save, Cut, Copy, Paste, Find, and others.
- Status Bar (bottom):** Shows the path `AlimentoController.java - Line 10, AlimentoController.java` and the status bar text "File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help".

## CadastroController.java

The screenshot shows the Eclipse IDE interface with the project 'jpa-entidad-api-4.0.1' selected. The 'CaptchaServlet.java' file is open in the editor. The code handles a POST request to '/captcha'. It checks if the parameter 'captcha' is null or empty. If it's empty, it generates a random string of length 5 and stores it in session attributes ('captcha', 'captcha\_md5'). It then forwards the request to 'index.jsp'. If 'captcha' is provided, it checks its length and md5 hash against session attributes. If they match, it forwards the request to 'index.jsp'. Otherwise, it sends an error response.

```
String cap = request.getParameter("captcha");
if (cap == null || cap.trim().isEmpty()) {
    request.setAttribute("captcha", "CAPA CREADO. Digite espacios numeros.");
    RequestDispatcher rd = request.getRequestDispatcher("index/captcha.jsp");
    rd.forward(request, response);
    return;
}

// ATRIBUTO RANDOM
cap = new String(5, 'a');

if (cap.length() != 5) {
    response.setAttribute("captcha", "CAPA DEBE TENER 5 DIGITOS.");
    RequestDispatcher rd = request.getRequestDispatcher("index/captcha.jsp");
    rd.forward(request, response);
    return;
}

Cill.setCOTP(cap);

// SE ENVIA AL INDEX
cil.setSession(request.getSession());
cil.setSessionAttribute("captcha", cil);
cil.setSessionAttribute("captcha_md5", cap);
cil.setSessionAttribute("idUsuario", cil.getId());
cil.setSessionAttribute("idSesion", cil.getId());

cil.sendRedirect("index.jsp");
```

This screenshot shows the same Eclipse IDE environment with the 'CaptchaServlet.java' file open. The code has been modified to handle a GET request to '/captcha'. It retrieves the session attribute 'captcha\_md5' and generates a random string of length 5. It then compares the provided 'captcha' with the stored MD5 hash. If they match, it sends a success response; otherwise, it sends an error response.

```
String captcha_md5 = request.getSession().getAttribute("captcha_md5").toString();
String cap = new String(5, 'a');

if (captcha_md5 != null) {
    if (cap.equals(captcha_md5)) {
        response.sendRedirect(request.getContextPath() + "/index/home.jsp");
    } else {
        response.sendError(request.getContextPath() + "/index/error.jsp");
    }
}
```

## CaptchaServelet.java

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (left):** Shows the project structure for "pacienteat-4.51-prueba" containing packages like "controlador", "model", "view", and "utils".
- Source Editor (center):** Displays the Java code for the "CaptchaServlet" class. The code includes imports for various Java packages, a constructor, and a doPost method. The doPost method contains logic for generating a random string, creating a BufferedImage, and saving it to a file.
- Outline View (right):** Shows the outline of the current file, including the package declaration, imports, and the doPost method.

## ConfiguracoesController.java

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (left):** Shows the project structure with packages like `com.apress.apresscoolapi`, `com.apress.apresscoolapi.controllers`, `com.apress.apresscoolapi.services`, and `com.apress.apresscoolapi.repositories`. It also lists various Java files such as `Application.java`, `ConfigurationsController.java`, `CustomerController.java`, `ImageController.java`, `ManagersController.java`, `ResponsesServiceController.java`, `UpgradesController.java`, and `UpgradesServiceController.java`.
- Code Editor (right):** Displays the `ConfigurationsController.java` file. The code implements a `ConfigurationsController` class that extends `HttpMessageNotModifiedHandler`. It handles requests for configuration details, including fetching specific configurations by ID and creating new ones. It also handles file uploads and manages session data.
- Top Bar:** Shows standard Eclipse menu items like File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help, and a tab for "Apache Tomcat 8.5".
- Bottom Status Bar:** Shows the current line number (154) and file name (index.html).

## EditarDadosController.java

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (left):** Shows the project structure for "processador-api-0.0.1-SNAPSHOT". It includes packages like "processador", "util", "log", "exception", "enums", "models", "controller", "repository", "service", and "utils". Under "controller", there are several controller classes: "AlunoController.java", "CapturaController.java", "ConfiguracaoController.java", "EnderecoController.java", "EmailController.java", "FotoController.java", "InscricaoController.java", "LogController.java", "MatriculaController.java", "RespostaController.java", "SaidaFotoController.java", "UploadImageController.java", and "UploadImageController.java".
  - File Structure:** The "controller" package contains:
    - AlunoController.java:** Contains logic for handling student requests, including methods like "salvarAluno" and "listarAlunos". It uses "ClientDAO" and "EmailService" to interact with the database and send emails.
    - CapturaController.java:** Handles image uploads, using "ImageUtil" and "ImageStorage" services.
    - ConfiguracaoController.java:** Manages configuration settings.
    - EnderecoController.java:** Handles address management.
    - EmailController.java:** Handles email sending.
    - FotoController.java:** Handles photo management.
    - InscricaoController.java:** Handles enrollment requests.
    - LogController.java:** Handles log management.
    - MatriculaController.java:** Handles matriculation requests.
    - RespostaController.java:** Handles response management.
    - SaidaFotoController.java:** Handles photo download requests.
    - UploadImageController.java:** Handles image upload requests.
- Source Editor (right):** Displays the content of the "Controller.java" file. The code is annotated with Javadoc comments explaining its purpose and parameters.
- Top Bar:** Shows the Eclipse menu bar (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help) and the Java EE perspective.
- Bottom Bar:** Shows the status bar with "Search (Ctrl+F)" and other system information.

## EmailSender.java

The screenshot shows the Eclipse IDE interface with the Java code for the `EmailSender` class. The code is annotated with comments explaining its purpose:

```
package controller;
// Classe utilizadora EmailSender
// Envia e-mail a destinatário
// Pode:
// - Enviar de e-mail via SMTP (Email)
// - Enviar de e-mail de recuperador de senha
// - inserir:
// - Enviar Any Recipient no E-mail (não suportado)
// - Apresentar e-mail de HTML simples

import java.util.Properties;
import javax.mail.MessagingException;
import javax.mail.PasswordAuthentication;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;

public class EmailSender {

    // CONSTRUCAO INICIAL DO E-MAIL
    private static final String EMAIL_DESTINATARIO = "filipe.silveira@infostimulus.com";
    private static final String SENHA_APP = "available@value";
    // I herda os aplicativos (não é a porta normal do e-mail)

    // MÉTODO PRINCIPAL DE ENVIO
    public static void enviarEmail(String destinatario, String assunto, String mensagemEmail) {
        // configurando AMT DO E-MAIL
        Properties props = new Properties();
        props.setProperty("mail.smtp.host", "smtp.infostimulus.com");
        props.put("mail.smtp.port", "587");
        props.put("mail.smtp.auth", "true");
        props.put("mail.smtp.starttls.enable", "true");
        props.put("mail.smtp.protocol", "TLS");
        Session session = Session.getInstance(props);
        Transport.send(message);
    }
}
```

The screenshot shows the Eclipse IDE interface with the same Java code for the `EmailSender` class. The code is annotated with comments explaining its purpose:

```
package controller;
// Classe utilizadora EmailSender
// Envia e-mail a destinatário
// Pode:
// - Enviar Any Recipient no E-mail (não suportado)
// - inserir:
// - Enviar Any Recipient no E-mail (não suportado)
// - Apresentar e-mail de HTML simples

import java.util.Properties;
import javax.mail.MessagingException;
import javax.mail.PasswordAuthentication;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;

public class EmailSender {

    // CONSTRUCAO INICIAL DO E-MAIL
    private static final String EMAIL_DESTINATARIO = "filipe.silveira@infostimulus.com";
    private static final String SENHA_APP = "available@value";
    // I herda os aplicativos (não é a porta normal do e-mail)

    // MÉTODO PRINCIPAL DE ENVIO
    public static void enviarEmail(String destinatario, String assunto, String mensagemEmail) {
        // configurando AMT DO E-MAIL
        Properties props = new Properties();
        props.setProperty("mail.smtp.host", "smtp.infostimulus.com");
        props.put("mail.smtp.port", "587");
        props.put("mail.smtp.auth", "true");
        props.put("mail.smtp.starttls.enable", "true");
        props.put("mail.smtp.protocol", "TLS");
        Session session = Session.getInstance(props);
        Transport.send(message);
    }
}
```

## FotoController.java

```
private final ClientesDAO dao = new ClientesDAO();
HttpURLConnection conexao = (HttpURLConnection) new URL(idCliente).openConnection();
if (conexao.getResponseCode() == null) {
    throw new IOException("Não foi possível obter o cliente");
}
String contentType = conexao.getContentType();
if (contentType == null) {
    contentType = "image/jpeg";
}

byte[] decodedbytes = CryptUtil.decrypt(conexao.getContent());
response.setContentType(contentType);
response.getOutputStream().write(decodedbytes);
}

catch (Exception e) {
    e.printStackTrace();
    response.setStatus(500);
    response.setErrorMessage("Erro ao processar foto");
}
}
}
```

ImageController.java

```
package controller;
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.util.List;
import model.Imagen;
import model.ImagenDAO;
public class ImagenController extends HttpServlet {
private final ImagenDAO dao = new ImagenDAO();
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    String idImagen = request.getParameter("id");
    if (idImagen == null) {
        response.sendError(HttpServletResponse.SC_BAD_REQUEST, "ID não informado.");
    }
    int id;
    try {
        id = Integer.parseInt(idImagen);
        if (id < 0) {
            response.sendError(HttpServletResponse.SC_BAD_REQUEST, "ID inválido.");
        }
    }
    catch (NumberFormatException e) {
        response.sendError(HttpServletResponse.SC_BAD_REQUEST, "ID inválido.");
    }
}
}
```

## LoginController.java

The screenshot shows the NetBeans IDE interface with the following details:

- File Bar:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Title Bar:** ApacheTomcat8 - Apache NetBeans (X:3).
- Project Tree:** Shows the project structure with several source packages and files, including `ControllerBase.java`, `Controller.java`, `CustomerController.java`, `ConfigurableController.java`, `EmployeeController.java`, `FileController.java`, `ImageController.java`, `LogicController.java`, `ManagerController.java`, `ReportController.java`, `ReportableController.java`, `UploadImageController.java`, and `UploadingController.java`.
- Code Editor:** The main window displays the `ImageController.java` file. The code is as follows:

```
1 package controllers;
2
3 import java.io.File;
4 import java.util.List;
5 import javax.servlet.http.HttpServletRequest;
6 import javax.servlet.http.HttpServletResponse;
7
8 import model.Client;
9 import modelDAO.ClientDAO;
10
11 public class ImageController extends BaseController {
12
13     @Override
14     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
15         String captionInput = request.getParameter("caption_input");
16         String captionName = (String) request.getParameter("caption_name").getCaption();
17
18         if (captionName == null || !captionName.equals(captionInput)) {
19             request.setAttribute("caption", "Caption incorrect. Termes nemnek.");
20             request.getRequestDispatcher("login/login.jsp").forward(request, response);
21         }
22
23         Client client = new Client();
24
25         ClientDAO dao = new ClientDAO();
26
27         client.setCaption(captionName);
28         client.setCaptionInput(captionInput);
29         client.setPhoto((byte[]) request.getAttribute("image"));
30
31         dao.create(client);
32
33         response.sendRedirect("index.jsp");
34     }
35 }
```

Screenshot of the Apache NetBeans IDE showing the code editor for the `LopeController.java` file. The code implements a login controller for a web application.

```

    package controller;
    ...
    import javax.servlet.http.HttpServletRequest;
    import javax.servlet.http.HttpServletResponse;
    import javax.servlet.http.HttpSession;
    ...
    public class LopeController {
        ...
        @Override
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
            HttpSession session = request.getSession();
            String email = request.getParameter("email");
            String password = request.getParameter("password");
            ...
            if (email != null & password != null) {
                ...
                session.setAttribute("username", email);
                session.setAttribute("password", password);
                ...
            }
        }
    }

```

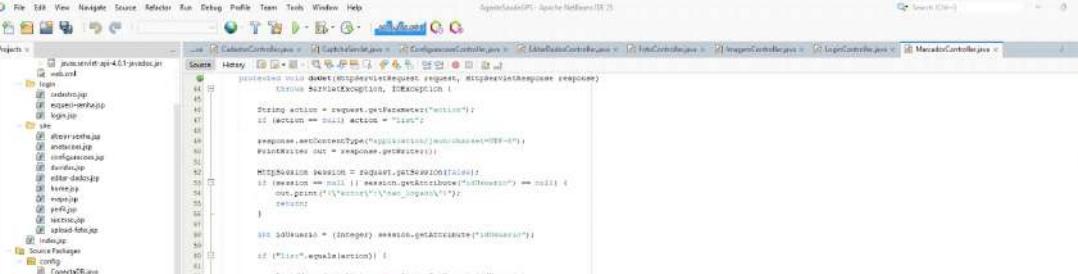
## MarcadorController.java

Screenshot of the Apache NetBeans IDE showing the code editor for the `MarcadorController.java` file. The code implements a marcador controller for a web application.

```

    package controller;
    ...
    import java.io.IOException;
    import java.util.ArrayList;
    import java.util.List;
    ...
    import javax.servlet.ServletException;
    import javax.servlet.annotation.WebServlet;
    import javax.servlet.http.HttpServlet;
    import javax.servlet.http.HttpServletRequest;
    import javax.servlet.http.HttpServletResponse;
    ...
    @WebServlet("/marcador")
    public class MarcadorController extends HttpServlet {
        ...
        private static MarcadorDAO mDAO = new MarcadorDAO();
        ...
        @Override
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
            ...
            String action = request.getParameter("action");
            ...
        }
    }

```



```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    String action = request.getParameter("action");
    if (action == null) action = "list";
    response.setContentType("application/json; charset=UTF-8");
    PrintWriter out = response.getWriter();
    HttpSession session = request.getSession(true);
    if (session.getAttribute("ultimoNota") == null) {
        out.print("{\"ultimo\": \"none\"}");
    } else {
        int ultimoNota = (Integer) session.getAttribute("ultimoNota");
        if ("list".equals(action)) {
            List lista = NotaListasPorOrigen(ultimoNota);
            Stringholder json = new Stringholder("[");

            for (int i = 0; i < lista.size(); i++) {
                Nota nota = lista.get(i);
                String notaStr = (nota.getNota() == null || nota.getNota().trim().isEmpty())
                        ? "" : nota.getNota();
                json.append(notaStr);
                json.append(",");
                json.append("\"origen\":");
                json.append("\"" + nota.getOrigen() + "\"").append(",");
                json.append("\"listado\":");
                json.append("[").append(nota.getLista().getNota());
                json.append(",");
                json.append("]").append(",");
                json.append("\"ultimo\":");
                json.append("[").append(nota.getUltimoNota());
                json.append(",");
                json.append("]");
            }
            if (G > lista.size() - 1) json.append("]");
        }
        json.append(")");
        out.print(json.toString());
    }
}
```

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (Left):** Shows the project structure with packages like `com.parcelservice.api-0.0.1-provider`, `com.parcelservice`, `com.parcelservice.controllers`, `com.parcelservice.entities`, `com.parcelservice.repositories`, `com.parcelservice.services`, `com.parcelservice.utils`, and `com.parcelservice.web`. It also lists various JAR files and configuration files.
- Code Editor (Center):** Displays the Java code for the `ActualizarNota` method in the `NotaController.java` file. The code handles requests for updating notes, validating input parameters, and updating the database.
- Toolbars and Menus (Top):** Standard Eclipse menus like File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help, and a toolbar with icons for Save, Undo, Redo, Cut, Copy, Paste, Find, and Refresh.
- Status Bar (Bottom):** Shows the path `NotaController.java [Line 10]`.

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (left):** Displays the project structure for a Java application named "processor-api-0.0.1-project". It includes packages like "com", "controller", "entity", "exception", "filter", "model", "repository", and "util". Specific files shown include "Controller.java", "ControllerBase.java", "ControllerException.java", "ControllerFilter.java", "ControllerModel.java", "ControllerRepository.java", "ControllerUtil.java", "Entity.java", "Exception.java", "Filter.java", "Model.java", "Repository.java", and "Util.java".
- Code Editor (right):** Shows the source code for a controller class, likely "Controller.java". The code implements the "Controller" interface and extends "ControllerBase". It handles requests for "/list" and "/list/{id}" using annotations like @RequestMapping and @PathVariable. It also handles exceptions and returns JSON responses.
- Top Bar:** Shows the standard Eclipse menu bar (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Task, Window, Help) and the Apache Felix JMX agent information.

## RecuperarSenhaController.java

The screenshot shows the Apache NetBeans IDE interface with the following details:

- File Bar:** File, Edit, View, Navigator, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Project Bar:** Projects (selected), CaptchaController, ConfigurationController, LoginController, ImageController, PageController, DomainController, RecuperarSenhaController.
- Toolbar:** Standard icons for file operations like Open, Save, Print, etc.
- Source Editor:** Displays Java code for a class named `LoginController.java`. The code handles user login logic, including password hashing, sending emails, and managing session attributes. It uses annotations like `@Controller`, `@RequestMapping`, and `@PostConstruct`.
- Project Explorer:** Shows the project structure with packages like `login`, `util`, `model`, and `Test Packages`. Specific files listed include `loginController.java`, `imageController.java`, `domainController.java`, `recuperarSenhaController.java`, and various configuration and service files.
- Properties View:** Shows properties for the selected `LoginController.java` file, including build path, source path, and deployment settings.
- Output View:** Shows the output of the build process.
- Help View:** Provides access to the Java API documentation.

## UploadFotoController.java

## UploadImageController.java

## Login.jsp

The screenshot shows the NetBeans IDE interface with the following details:

- Title Bar:** Apache NetBeans IDE - AgentSelvadorIPS - Apache NetBeans IDE
- Toolbar:** File, Edit, View, Navigator, Source, Refactor, Run, Debug, Profile, Tools, Window, Help.
- Project Explorer:** Shows a Java project structure with packages like `br.com.silva.sistema`, `br.com.silva.sistema.controllers`, `br.com.silva.sistema.models`, and `br.com.silva.sistema.utils`. A file named `LoginController.java` is selected.
- Code Editor:** Displays the Java code for the `LoginController`. The code handles user login logic, including cookie management and session creation. It includes annotations like `@Controller`, `@RequestMapping`, and `@SessionAttribute`.
- Toolbars:** Standard NetBeans toolbars for navigation and code editing.

The screenshot shows the Apache NetBeans IDE interface with the following details:

- Title Bar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Toolbar:** New Open Save Run Clean Build Deploy Find Replace Go
- Project Explorer (Left):** Shows the project structure for "AgentServerGUISession". It includes packages like AgentServerGUISession, WebPages, META-INF, and various controllers (e.g., AgentController.java, LoginController.java, LogoutController.java, MainController.java, RequestController.java, SessionController.java, UploadImageController.java, UploadSessionController.java). It also lists files like index.jsp, config, and various XML files.
- Code Editor (Right):** Displays the Java code for "AgentServerGUISessionController.java". The code handles session management, including session creation, login, logout, and session destruction. It uses annotations like @SessionController, @SessionAttribute, and @PostConstruct. The code includes imports for javax.faces.bean, javax.faces.context, javax.faces.event, javax.faces.model, javax.faces.view, javax.servlet.http, javax.servlet.http.HttpSession, and javax.servlet.http.Part.

Screenshot of the NetBeans IDE showing the source code for the 'LoginController.java' file. The code is a CSS file (logins.css) containing styles for various UI components like buttons, inputs, and labels.

```
Project: AgentesGeolP
File: logins.css
Source: History: Logins.css - AgentesGeolP - Apache NetBeans (10.25)
1 /* Botões de login e cancelamento */
2 .botao {
3     width: 100px;
4     height: 30px;
5     border-radius: 10px;
6     background-color: #007bff;
7     color: white;
8     font-weight: bold;
9     font-size: 14px;
10    padding: 5px;
11    margin-bottom: 10px;
12 }
13 
14 .cancelar {
15     background-color: #d9e1f2;
16     border: none;
17     color: black;
18     cursor: pointer;
19 }
20 
21 .cancelar:active {
22     background-color: #cde5f6;
23 }
24 
25 .cancelar:disabled {
26     background-color: #cccccc;
27     color: gray;
28     cursor: not-allowed;
29 }
30 
31 /* Formulário de login/cadastro */
32 .formularios {
33     width: 300px;
34     margin: auto;
35     text-align: center;
36     border: 1px solid #ccc;
37     padding: 10px;
38     border-radius: 10px;
39     background-color: #fff;
40 }
41 
42 .formularios input {
43     width: 100px;
44     height: 30px;
45     border: 1px solid #ccc;
46     border-radius: 5px;
47     padding: 5px;
48     margin-bottom: 10px;
49 }
50 
51 .formularios input[type="text"] {
52     width: 100px;
53     height: 30px;
54     border: 1px solid #ccc;
55     border-radius: 5px;
56     padding: 5px;
57     margin-bottom: 10px;
58 }
59 
60 .formularios input[type="password"] {
61     width: 100px;
62     height: 30px;
63     border: 1px solid #ccc;
64     border-radius: 5px;
65     padding: 5px;
66     margin-bottom: 10px;
67 }
68 
69 .formularios input[type="checkbox"] {
70     width: 100px;
71     height: 30px;
72     border: 1px solid #ccc;
73     border-radius: 5px;
74     padding: 5px;
75     margin-bottom: 10px;
76 }
77 
78 .formularios input[type="radio"] {
79     width: 100px;
80     height: 30px;
81     border: 1px solid #ccc;
82     border-radius: 5px;
83     padding: 5px;
84     margin-bottom: 10px;
85 }
86 
87 .formularios input[type="button"] {
88     width: 100px;
89     height: 30px;
90     border: 1px solid #ccc;
91     border-radius: 5px;
92     padding: 5px;
93     background-color: #007bff;
94     color: white;
95     font-weight: bold;
96     font-size: 14px;
97     margin-top: 10px;
98     margin-bottom: 10px;
99 }
100 
101 .formularios input[type="button"]:active {
102     background-color: #cde5f6;
103 }
104 
105 .formularios input[type="button":disabled] {
106     background-color: #cccccc;
107     color: gray;
108     cursor: not-allowed;
109 }
110 
111 /* Botão de enviar formulário */
112 .enviar {
113     width: 100px;
114     height: 30px;
115     border: 1px solid #ccc;
116     border-radius: 5px;
117     padding: 5px;
118     background-color: #007bff;
119     color: white;
120     font-weight: bold;
121     font-size: 14px;
122     margin-top: 10px;
123 }
124 
125 .enviar:active {
126     background-color: #cde5f6;
127 }
128 
129 .enviar:disabled {
130     background-color: #cccccc;
131     color: gray;
132     cursor: not-allowed;
133 }
134 
135 /* Formulário de cadastro */
136 .cadastro {
137     width: 300px;
138     margin: auto;
139     text-align: center;
140     border: 1px solid #ccc;
141     padding: 10px;
142     border-radius: 10px;
143     background-color: #fff;
144 }
145 
146 .cadastro input {
147     width: 100px;
148     height: 30px;
149     border: 1px solid #ccc;
150     border-radius: 5px;
151     padding: 5px;
152     margin-bottom: 10px;
153 }
154 
155 .cadastro input[type="text"] {
156     width: 100px;
157     height: 30px;
158     border: 1px solid #ccc;
159     border-radius: 5px;
160     padding: 5px;
161     margin-bottom: 10px;
162 }
163 
164 .cadastro input[type="password"] {
165     width: 100px;
166     height: 30px;
167     border: 1px solid #ccc;
168     border-radius: 5px;
169     padding: 5px;
170     margin-bottom: 10px;
171 }
172 
173 .cadastro input[type="checkbox"] {
174     width: 100px;
175     height: 30px;
176     border: 1px solid #ccc;
177     border-radius: 5px;
178     padding: 5px;
179     margin-bottom: 10px;
180 }
181 
182 .cadastro input[type="radio"] {
183     width: 100px;
184     height: 30px;
185     border: 1px solid #ccc;
186     border-radius: 5px;
187     padding: 5px;
188     margin-bottom: 10px;
189 }
190 
191 .cadastro input[type="button"] {
192     width: 100px;
193     height: 30px;
194     border: 1px solid #ccc;
195     border-radius: 5px;
196     padding: 5px;
197     background-color: #007bff;
198     color: white;
199     font-weight: bold;
200     font-size: 14px;
201     margin-top: 10px;
202 }
203 
204 .cadastro input[type="button"]:active {
205     background-color: #cde5f6;
206 }
207 
208 .cadastro input[type="button":disabled] {
209     background-color: #cccccc;
210     color: gray;
211     cursor: not-allowed;
212 }
213 
214 /* Formulário de login com máscara */
215 .login-mascara {
216     width: 300px;
217     margin: auto;
218     text-align: center;
219     border: 1px solid #ccc;
220     padding: 10px;
221     border-radius: 10px;
222     background-color: #fff;
223 }
224 
225 .login-mascara input {
226     width: 100px;
227     height: 30px;
228     border: 1px solid #ccc;
229     border-radius: 5px;
230     padding: 5px;
231     margin-bottom: 10px;
232 }
233 
234 .login-mascara input[type="text"] {
235     width: 100px;
236     height: 30px;
237     border: 1px solid #ccc;
238     border-radius: 5px;
239     padding: 5px;
240     margin-bottom: 10px;
241 }
242 
243 .login-mascara input[type="password"] {
244     width: 100px;
245     height: 30px;
246     border: 1px solid #ccc;
247     border-radius: 5px;
248     padding: 5px;
249     margin-bottom: 10px;
250 }
251 
252 .login-mascara input[type="checkbox"] {
253     width: 100px;
254     height: 30px;
255     border: 1px solid #ccc;
256     border-radius: 5px;
257     padding: 5px;
258     margin-bottom: 10px;
259 }
260 
261 .login-mascara input[type="radio"] {
262     width: 100px;
263     height: 30px;
264     border: 1px solid #ccc;
265     border-radius: 5px;
266     padding: 5px;
267     margin-bottom: 10px;
268 }
269 
270 .login-mascara input[type="button"] {
271     width: 100px;
272     height: 30px;
273     border: 1px solid #ccc;
274     border-radius: 5px;
275     padding: 5px;
276     background-color: #007bff;
277     color: white;
278     font-weight: bold;
279     font-size: 14px;
280     margin-top: 10px;
281 }
282 
283 .login-mascara input[type="button"]:active {
284     background-color: #cde5f6;
285 }
286 
287 .login-mascara input[type="button":disabled] {
288     background-color: #cccccc;
289     color: gray;
290     cursor: not-allowed;
291 }
292 
293 /* Formulário de login com máscara e placeholder */
294 .login-mascara-placeholder {
295     width: 300px;
296     margin: auto;
297     text-align: center;
298     border: 1px solid #ccc;
299     padding: 10px;
300     border-radius: 10px;
301     background-color: #fff;
302 }
303 
304 .login-mascara-placeholder input {
305     width: 100px;
306     height: 30px;
307     border: 1px solid #ccc;
308     border-radius: 5px;
309     padding: 5px;
310     margin-bottom: 10px;
311 }
312 
313 .login-mascara-placeholder input[type="text"] {
314     width: 100px;
315     height: 30px;
316     border: 1px solid #ccc;
317     border-radius: 5px;
318     padding: 5px;
319     margin-bottom: 10px;
320 }
321 
322 .login-mascara-placeholder input[type="password"] {
323     width: 100px;
324     height: 30px;
325     border: 1px solid #ccc;
326     border-radius: 5px;
327     padding: 5px;
328     margin-bottom: 10px;
329 }
330 
331 .login-mascara-placeholder input[type="checkbox"] {
332     width: 100px;
333     height: 30px;
334     border: 1px solid #ccc;
335     border-radius: 5px;
336     padding: 5px;
337     margin-bottom: 10px;
338 }
339 
340 .login-mascara-placeholder input[type="radio"] {
341     width: 100px;
342     height: 30px;
343     border: 1px solid #ccc;
344     border-radius: 5px;
345     padding: 5px;
346     margin-bottom: 10px;
347 }
348 
349 .login-mascara-placeholder input[type="button"] {
350     width: 100px;
351     height: 30px;
352     border: 1px solid #ccc;
353     border-radius: 5px;
354     padding: 5px;
355     background-color: #007bff;
356     color: white;
357     font-weight: bold;
358     font-size: 14px;
359     margin-top: 10px;
360 }
361 
362 .login-mascara-placeholder input[type="button"]:active {
363     background-color: #cde5f6;
364 }
365 
366 .login-mascara-placeholder input[type="button":disabled] {
367     background-color: #cccccc;
368     color: gray;
369     cursor: not-allowed;
370 }
371 
372 /* Formulário de login com máscara e placeholder e placeholder para telefone */
373 .login-mascara-placeholder-telefone {
374     width: 300px;
375     margin: auto;
376     text-align: center;
377     border: 1px solid #ccc;
378     padding: 10px;
379     border-radius: 10px;
380     background-color: #fff;
381 }
382 
383 .login-mascara-placeholder-telefone input {
384     width: 100px;
385     height: 30px;
386     border: 1px solid #ccc;
387     border-radius: 5px;
388     padding: 5px;
389     margin-bottom: 10px;
390 }
391 
392 .login-mascara-placeholder-telefone input[type="text"] {
393     width: 100px;
394     height: 30px;
395     border: 1px solid #ccc;
396     border-radius: 5px;
397     padding: 5px;
398     margin-bottom: 10px;
399 }
400 
401 .login-mascara-placeholder-telefone input[type="password"] {
402     width: 100px;
403     height: 30px;
404     border: 1px solid #ccc;
405     border-radius: 5px;
406     padding: 5px;
407     margin-bottom: 10px;
408 }
409 
410 .login-mascara-placeholder-telefone input[type="checkbox"] {
411     width: 100px;
412     height: 30px;
413     border: 1px solid #ccc;
414     border-radius: 5px;
415     padding: 5px;
416     margin-bottom: 10px;
417 }
418 
419 .login-mascara-placeholder-telefone input[type="radio"] {
420     width: 100px;
421     height: 30px;
422     border: 1px solid #ccc;
423     border-radius: 5px;
424     padding: 5px;
425     margin-bottom: 10px;
426 }
427 
428 .login-mascara-placeholder-telefone input[type="button"] {
429     width: 100px;
430     height: 30px;
431     border: 1px solid #ccc;
432     border-radius: 5px;
433     padding: 5px;
434     background-color: #007bff;
435     color: white;
436     font-weight: bold;
437     font-size: 14px;
438     margin-top: 10px;
439 }
440 
441 .login-mascara-placeholder-telefone input[type="button"]:active {
442     background-color: #cde5f6;
443 }
444 
445 .login-mascara-placeholder-telefone input[type="button":disabled] {
446     background-color: #cccccc;
447     color: gray;
448     cursor: not-allowed;
449 }
450 
451 /* Formulário de login com máscara e placeholder e placeholder para telefone e placeholder para CPF */
452 .login-mascara-placeholder-telefone-cpf {
453     width: 300px;
454     margin: auto;
455     text-align: center;
456     border: 1px solid #ccc;
457     padding: 10px;
458     border-radius: 10px;
459     background-color: #fff;
460 }
461 
462 .login-mascara-placeholder-telefone-cpf input {
463     width: 100px;
464     height: 30px;
465     border: 1px solid #ccc;
466     border-radius: 5px;
467     padding: 5px;
468     margin-bottom: 10px;
469 }
470 
471 .login-mascara-placeholder-telefone-cpf input[type="text"] {
472     width: 100px;
473     height: 30px;
474     border: 1px solid #ccc;
475     border-radius: 5px;
476     padding: 5px;
477     margin-bottom: 10px;
478 }
479 
480 .login-mascara-placeholder-telefone-cpf input[type="password"] {
481     width: 100px;
482     height: 30px;
483     border: 1px solid #ccc;
484     border-radius: 5px;
485     padding: 5px;
486     margin-bottom: 10px;
487 }
488 
489 .login-mascara-placeholder-telefone-cpf input[type="checkbox"] {
490     width: 100px;
491     height: 30px;
492     border: 1px solid #ccc;
493     border-radius: 5px;
494     padding: 5px;
495     margin-bottom: 10px;
496 }
497 
498 .login-mascara-placeholder-telefone-cpf input[type="radio"] {
499     width: 100px;
500     height: 30px;
501     border: 1px solid #ccc;
502     border-radius: 5px;
503     padding: 5px;
504     margin-bottom: 10px;
505 }
506 
507 .login-mascara-placeholder-telefone-cpf input[type="button"] {
508     width: 100px;
509     height: 30px;
510     border: 1px solid #ccc;
511     border-radius: 5px;
512     padding: 5px;
513     background-color: #007bff;
514     color: white;
515     font-weight: bold;
516     font-size: 14px;
517     margin-top: 10px;
518 }
519 
520 .login-mascara-placeholder-telefone-cpf input[type="button"]:active {
521     background-color: #cde5f6;
522 }
523 
524 .login-mascara-placeholder-telefone-cpf input[type="button":disabled] {
525     background-color: #cccccc;
526     color: gray;
527     cursor: not-allowed;
528 }
529 
```

Screenshot of the NetBeans IDE showing the source code for the 'LoginController.java' file. The code is a Java class containing methods for handling user authentication and registration.

```
Project: AgentesGeolP
File: LoginController.java
Source: History: LoginController.java - AgentesGeolP - Apache NetBeans (10.25)
1 package br.com.agentesgeol;
2 
3 import br.com.agentesgeol.model.Cliente;
4 import br.com.agentesgeol.model.Imagem;
5 import br.com.agentesgeol.model.Usuario;
6 import br.com.agentesgeol.model.UsuarioDAO;
7 import br.com.agentesgeol.model.UsuarioDAOImpl;
8 import br.com.agentesgeol.util.Mensagem;
9 import br.com.agentesgeol.util.MensagemDAO;
10 import br.com.agentesgeol.util.MensagemDAOImpl;
11 import java.io.IOException;
12 import javax.servlet.ServletException;
13 import javax.servlet.annotation.WebServlet;
14 import javax.servlet.http.HttpServlet;
15 import javax.servlet.http.HttpServletRequest;
16 import javax.servlet.http.HttpServletResponse;
17 import javax.servlet.http.HttpSession;
18 
19 /**
20 * Servlet implementation class LoginController
21 */
22 @WebServlet("/LoginController")
23 public class LoginController extends HttpServlet {
24 
25     protected void processRequest(HttpServletRequest request, HttpServletResponse response)
26             throws ServletException, IOException {
27         response.setContentType("text/html;charset=UTF-8");
28 
29         String email = request.getParameter("email");
30         String senha = request.getParameter("senha");
31 
32         HttpSession session = request.getSession();
33 
34         UsuarioDAO usuarioDAO = new UsuarioDAOImpl();
35         Usuario usuario = usuarioDAO.getUsuario(email);
36 
37         if (usuario != null) {
38             if (senha.equals(usuario.getSenha())) {
39                 session.setAttribute("usuarioLogado", usuario);
40 
41                 response.sendRedirect("index.jsp");
42             } else {
43                 Mensagem.setMensagem("Senha incorreta!");
44                 response.sendRedirect("login.jsp");
45             }
46         } else {
47             Mensagem.setMensagem("Email não encontrado!");
48             response.sendRedirect("login.jsp");
49         }
50     }
51 
52     // 
53     //   

54     // HTTP GET
55     /* Handles requests method=GET */
56     protected void doGet(HttpServletRequest request, HttpServletResponse response)
57             throws ServletException, IOException {
58         processRequest(request, response);
59     }
60 
61     // HTTP POST
62     /* Handles requests method=POST */
63     protected void doPost(HttpServletRequest request, HttpServletResponse response)
64             throws ServletException, IOException {
65         processRequest(request, response);
66     }
67 
68     // HTTP HEAD
69     /* Handles requests method=HEAD */
70     protected void doHead(HttpServletRequest request, HttpServletResponse response)
71             throws ServletException, IOException {
72         processRequest(request, response);
73     }
74     // 
75 }
76 
```

The screenshot shows the NetBeans IDE interface with the following details:

- Project Tree:** The project is named "ApacheShiroJava". It contains a "Web Pages" folder with "index.jsp" and "index2.jsp". There are also "src" and "Source Packages" sections. Under "Source Packages", there are several packages: "com", "com.controller", "com.model", and "com.util".
- Code Editor:** The current file being edited is "ImageController.java". The code is as follows:

```
// Atualiza a imagem quando o nome é alterado
function atualizarLogo() {
    var logo = document.getElementById("logo");
    if (document.body.classList.contains("logon")) {
        logo.src = "#!/img/logo.png";
    } else {
        logo.src = "#!/img/logo2.png";
    }
}

// Adiciona a classe de fundo da home
function atualizarHome() {
    var home = document.getElementsByClassName("logon");
    home.classList.add("logon");
}

// Adiciona a classe de fundo da login
function atualizarLogin() {
    document.getElementsByClassName("logon");
    var login = document.getElementsByClassName("logon");
    login.classList.add("logon");
}
```

Cadastro.jsp

The screenshot shows the NetBeans IDE interface with the following details:

- Project Tree:** The project is named "AgenteGesGPS". It contains several modules:
  - Web Pages
  - META-INF
  - WEB-INF
  - src
  - ejb
  - entity
  - faces
  - java
  - model
  - resources
  - tests
  - web
- Code Editor:** The current file is "Home.java" under the "faces" module. The code implements the "View" interface and defines a constructor and a method "listar".

```
public class Home extends View {    public Home() {        // Construtor para inicializar o formulário        // ...    }    public void listar() {        // Código para listar os dados da base de dados        // ...    }}
```
- Toolbar:** Standard NetBeans toolbar with icons for file operations, search, and project navigation.
- Status Bar:** Shows the current file path ("AgenteGesGPS - Apache Geronimo (J2EE)"), the line number (17), and the column number (84).



The screenshot shows the NetBeans IDE interface with the following details:

- File**, **Edit**, **View**, **Navigator**, **Source**, **Advertis**, **Run**, **Debug**, **Profile**, **Team**, **Tools**, **Window**, **Help** menu items.
- Project tree on the left:
  - AgentsStateGPS**: Web Page, META-INF, resources, controllers, models, views.
    - views**: index.jsp, login.jsp, loginError.jsp, loginSuccess.jsp, logout.jsp, register.jsp, registerSuccess.jsp, userEdit.jsp, userEditSuccess.jsp, userList.jsp, userNew.jsp, userNewSuccess.jsp.
  - Source Packages**:
    - com: Controller, Model, Util.
    - com.agentsstategps: AgentsController.java, AgentsControllerTest.java, AgentsModel.java, AgentsUtil.java, AgentsView.java, AgentsViewTest.java, AgentsViewUtil.java, AgentsViewUtilTest.java.
    - com.agentsstategps.controllers: AgentsController.java, AgentsControllerTest.java, AgentsControllerUtil.java, AgentsControllerUtilTest.java.
    - com.agentsstategps.controllers.concretes: AgentsControllerConcrete.java, AgentsControllerConcreteTest.java.
    - com.agentsstategps.controllers.filters: AgentsFilterController.java, AgentsFilterControllerTest.java.
    - com.agentsstategps.controllers.listeners: AgentsListenerController.java, AgentsListenerControllerTest.java.
    - com.agentsstategps.controllers.util: AgentsControllerUtil.java, AgentsControllerUtilTest.java.
    - com.agentsstategps.model: AgentsModel.java, AgentsModelTest.java.
    - com.agentsstategps.util: AgentsUtil.java, AgentsUtilTest.java.
- Code editor window showing the `index.jsp` file content.
- Toolbar at the top with icons for file operations, search, and help.
- Bottom status bar showing "15039 INS".

esqueci-senha.jsp

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (Left):** Shows the project structure with several Java files under the "controller" package, such as LogonController.java, LoginController.java, and UploadImageController.java.
- Source Editor (Center):** Displays the Java code for LogonController.java. The code includes annotations like @Controller, @RequestMapping, and @PostConstruct, along with CSS-like styles defined in the code itself using the @Style annotation.
- Top Bar:** Shows the standard Eclipse menu items (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help) and the current workspace path (AgentsServer/AgentsServer (1)).
- Bottom Status Bar:** Shows the status "Ready" and the number of errors (0).

```
/* Estilos para los botones */
button {
    background-color: #007bff;
    color: white;
    border: none;
    padding: 10px 20px;
    border-radius: 5px;
    font-size: 16px;
    cursor: pointer;
}

button:hover {
    background-color: #0056b3;
    border: 2px solid #0056b3;
}

button:active {
    background-color: #004d99;
    border: 2px solid #004d99;
}

button:disabled {
    background-color: #cccccc;
    border: 2px solid #cccccc;
    opacity: 0.5;
}

/* Estilos para los inputs */
input[type="text"] {
    width: 100px;
    padding: 5px;
    border-radius: 5px;
    border: 1px solid #ccc;
    font-size: 14px;
}

input[type="text"]:focus {
    border: 2px solid #007bff;
    outline: none;
}

input[type="password"] {
    width: 100px;
    padding: 5px;
    border-radius: 5px;
    border: 1px solid #ccc;
    font-size: 14px;
}

input[type="password"]:focus {
    border: 2px solid #007bff;
    outline: none;
}

input[type="checkbox"] {
    width: 15px;
    height: 15px;
    border: 1px solid #ccc;
    border-radius: 50%;
    background-color: #fff;
    margin-right: 10px;
}

input[type="checkbox"]:checked {
    border: 2px solid #007bff;
    background-color: #007bff;
    color: white;
}
```



The screenshot shows the NetBeans IDE interface with the following details:

- Project Tree:** The project is named "AgentGradeGPS". It contains several packages:
  - Web Pages:** Contains files like index.jsp, login.jsp, register.jsp, forgot.jsp, and others.
  - META-INF:** Contains manifest.mf.
  - controller:** Contains LogController.java.
  - model:** Contains Client.java, Image.java, and Produto.java.
  - modelDAO:** Contains ClientDAO.java, ImageDAO.java, and ProdutoDAO.java.
  - util:** Contains EncryptUtil.java.
- Code Editor:** The current file is "LogController.java" under the "controller" package. The code implements the "Controller" interface and handles user authentication logic.
- Toolbar:** Standard NetBeans toolbar with icons for file operations, search, and execution.
- Status Bar:** Shows the message "AgentGradeGPS displayed".

site.jsp

The screenshot shows the NetBeans IDE interface with the following details:

- Project Tree:** Shows the project structure with several Java files under "Sources" and "Web Pages".
- Code Editor:** Displays the content of the file `index.jsp`. The code includes JSP tags, Java scriptlets, and a large block of Java code for handling user login.
- Toolbars:** Standard NetBeans toolbars for file operations, code navigation, and project management.
- Status Bar:** Shows the path "index.jsp - Apache NetBeans (38,75)" and the status "Search (Ctrl+F)".

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%
    // Recupera o contexto da aplicação para construção do URL absoluto.
    String ctx = request.getServletContext();
    // Redireciona ao usuário para autenticação.
    // Caso não exista, faz o redirecionamento imediato para a tela de login.
    IntegerultimoId = (Integer) session.getAttribute("ultimoId");
    if (ultimoId == null) {
        response.sendRedirect(ctx + "/login/login.jsp");
        return;
    }
    // Aqui recupera o número de sessões respondidas por exibir no topo.
    // Se não for nenhuma, o nome recém é exibido com parêntesis.
    String novoCookie = "none";
    HttpSessionCookie cookie = request.getAttribute("cookie");
    if (cookie != null) {
        for (Cookie c : cookie.getCookieList()) {
            if (c.getName().equals("ultimoId")) {
                novoCookie = c.getValue();
            }
        }
    }
    %>
<%
    // Criação da URL.
    final String url = "http://"+request.getServerName()+
    ":"+request.getServerPort()+
    "/"+request.getContextPath()+
    "/"+ultimoId+".jsp";
    // Verifica se existe uma sessão.
    if (novoCookie.equals("none")) {
        // Redireciona para a página de login.
        response.sendRedirect(url);
    } else {
        // Criação da sessão.
        session.setAttribute("ultimoId", ultimoId);
        // Redireciona para a página de destino.
        response.sendRedirect(url);
    }
%>
```

```
Project : AgentesGPI - Apache NetBeans IDE 12.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
LogController.java MarcaController.java ResponseFormatController.java UploadImageController.java login.jsp alert-senha.jsp request-senha.jsp

Source History Search (Ctrl+F) X


    ...
    --color: #000;
    --font-size: 1em;
    --font-weight: bold;
    --text-align: center;
    --text-decoration: none;
    --background-color: #fff;
    --border: 1px solid #ccc;
    --border-radius: 5px;
    --outline: none;
    --padding: 10px;
    --margin: 10px auto;
    --width: fit-content;
    --height: fit-content;
}

/* Botão de alternância de tema, quando no modo escuro, alterna automaticamente para o modo escuro */
button[aria-label="Switch theme"] {
    background-color: var(--bg);
    color: var(--text);
    font-family: "Sergo UI";
    transition: .2s;
}

/* Item de lista com scroll */
ul li {
    position: relative;
    width: 100px;
    height: 100px;
    background: linear-gradient(45deg, transparent, transparent 50%, #000 50%, #000 100%); /* Fazendo a cor só aparecer na metade direita da barra de rolagem */
    border-radius: 10px;
    overflow: hidden;
    margin-bottom: 10px;
}

/* Botão principal de pagina */
button[aria-label="Home"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    cursor: pointer;
    transition: .2s;
}

/* Botão de alternância de tema */
button[aria-label="Switch theme"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    border-radius: 5px;
    padding: 5px;
    margin: 10px auto;
    width: fit-content;
    height: fit-content;
}

/* Botão de alternância de tema, quando no modo escuro, alterna automaticamente para o modo escuro */
button[aria-label="Switch theme"] {
    background-color: var(--bg);
    color: var(--text);
    font-family: "Sergo UI";
    transition: .2s;
}

/* Item de lista com scroll */
ul li {
    position: relative;
    width: 100px;
    height: 100px;
    background: linear-gradient(45deg, transparent, transparent 50%, #000 50%, #000 100%); /* Fazendo a cor só aparecer na metade direita da barra de rolagem */
    border-radius: 10px;
    overflow: hidden;
    margin-bottom: 10px;
}

/* Botão principal de pagina */
button[aria-label="Home"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    cursor: pointer;
    transition: .2s;
}

/* Botão de alternância de tema */
button[aria-label="Switch theme"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    border-radius: 5px;
    padding: 5px;
    margin: 10px auto;
    width: fit-content;
    height: fit-content;
}


```

```
Project : AgentesGPI - Apache NetBeans IDE 12.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
LogController.java MarcaController.java ResponseFormatController.java UploadImageController.java login.jsp alert-senha.jsp request-senha.jsp

Source History Search (Ctrl+F) X


    ...
    --color: #000;
    --font-size: 1em;
    --font-weight: bold;
    --text-align: center;
    --text-decoration: none;
    --background-color: #fff;
    --border: 1px solid #ccc;
    --border-radius: 5px;
    --outline: none;
    --padding: 10px;
    --margin: 10px auto;
    --width: fit-content;
    --height: fit-content;
}

/* Botão de alternância de tema, quando no modo escuro, alterna automaticamente para o modo escuro */
button[aria-label="Switch theme"] {
    background-color: var(--bg);
    color: var(--text);
    font-family: "Sergo UI";
    transition: .2s;
}

/* Item de lista com scroll */
ul li {
    position: relative;
    width: 100px;
    height: 100px;
    background: linear-gradient(45deg, transparent, transparent 50%, #000 50%, #000 100%); /* Fazendo a cor só aparecer na metade direita da barra de rolagem */
    border-radius: 10px;
    overflow: hidden;
    margin-bottom: 10px;
}

/* Botão principal de pagina */
button[aria-label="Home"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    cursor: pointer;
    transition: .2s;
}

/* Botão de alternância de tema */
button[aria-label="Switch theme"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    border-radius: 5px;
    padding: 5px;
    margin: 10px auto;
    width: fit-content;
    height: fit-content;
}

/* Botão de alternância de tema, quando no modo escuro, alterna automaticamente para o modo escuro */
button[aria-label="Switch theme"] {
    background-color: var(--bg);
    color: var(--text);
    font-family: "Sergo UI";
    transition: .2s;
}

/* Item de lista com scroll */
ul li {
    position: relative;
    width: 100px;
    height: 100px;
    background: linear-gradient(45deg, transparent, transparent 50%, #000 50%, #000 100%); /* Fazendo a cor só aparecer na metade direita da barra de rolagem */
    border-radius: 10px;
    overflow: hidden;
    margin-bottom: 10px;
}

/* Botão principal de pagina */
button[aria-label="Home"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    cursor: pointer;
    transition: .2s;
}

/* Botão de alternância de tema */
button[aria-label="Switch theme"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    border-radius: 5px;
    padding: 5px;
    margin: 10px auto;
    width: fit-content;
    height: fit-content;
}


```

```
Project : AgentesGPI - Apache NetBeans IDE 12.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
LogController.java MarcaController.java ResponseFormatController.java UploadImageController.java login.jsp alert-senha.jsp request-senha.jsp

Source History Search (Ctrl+F) X


    ...
    --color: #000;
    --font-size: 1em;
    --font-weight: bold;
    --text-align: center;
    --text-decoration: none;
    --background-color: #fff;
    --border: 1px solid #ccc;
    --border-radius: 5px;
    --outline: none;
    --padding: 10px;
    --margin: 10px auto;
    --width: fit-content;
    --height: fit-content;
}

/* Botão de alternância de tema, quando no modo escuro, alterna automaticamente para o modo escuro */
button[aria-label="Switch theme"] {
    background-color: var(--bg);
    color: var(--text);
    font-family: "Sergo UI";
    transition: .2s;
}

/* Item de lista com scroll */
ul li {
    position: relative;
    width: 100px;
    height: 100px;
    background: linear-gradient(45deg, transparent, transparent 50%, #000 50%, #000 100%); /* Fazendo a cor só aparecer na metade direita da barra de rolagem */
    border-radius: 10px;
    overflow: hidden;
    margin-bottom: 10px;
}

/* Botão principal de pagina */
button[aria-label="Home"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    cursor: pointer;
    transition: .2s;
}

/* Botão de alternância de tema */
button[aria-label="Switch theme"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    border-radius: 5px;
    padding: 5px;
    margin: 10px auto;
    width: fit-content;
    height: fit-content;
}

/* Botão de alternância de tema, quando no modo escuro, alterna automaticamente para o modo escuro */
button[aria-label="Switch theme"] {
    background-color: var(--bg);
    color: var(--text);
    font-family: "Sergo UI";
    transition: .2s;
}

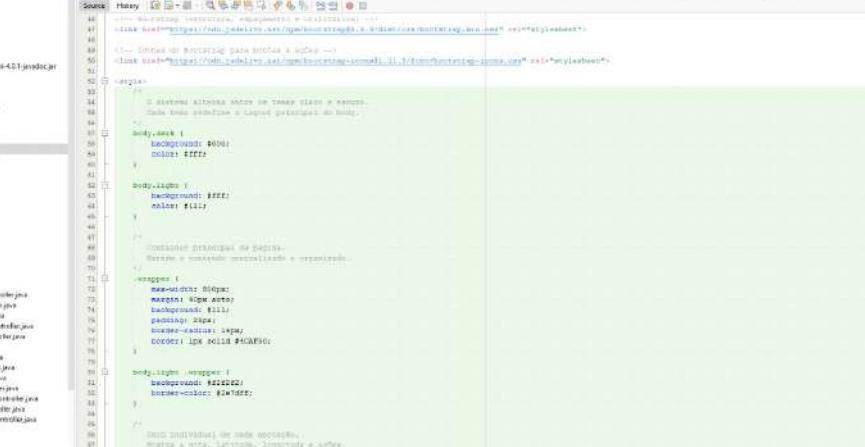
/* Item de lista com scroll */
ul li {
    position: relative;
    width: 100px;
    height: 100px;
    background: linear-gradient(45deg, transparent, transparent 50%, #000 50%, #000 100%); /* Fazendo a cor só aparecer na metade direita da barra de rolagem */
    border-radius: 10px;
    overflow: hidden;
    margin-bottom: 10px;
}

/* Botão principal de pagina */
button[aria-label="Home"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    cursor: pointer;
    transition: .2s;
}

/* Botão de alternância de tema */
button[aria-label="Switch theme"] {
    background-color: transparent;
    border: none;
    color: var(--text);
    border-radius: 5px;
    padding: 5px;
    margin: 10px auto;
    width: fit-content;
    height: fit-content;
}


```

anotacoes.jsp



The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (left):** Displays the "ApacheGeodeJava" project structure. It includes several Java files under "src/main/java" such as "index.java", "Client.java", "Image.java", "ImageController.java", "ImageDAO.java", "ImageDetailController.java", "ImageDetail.java", "ImageListController.java", "ImageList.java", "ImageUploadController.java", "ImageUpload.java", "LoginController.java", "LogoutController.java", "Logout.java", "MainController.java", "Main.java", "UserController.java", and "UserController.java". There are also configuration files like "application.properties" and "logback.xml".
- Code Editor (right):** Shows the content of the "index.css" file. The code defines a CSS style sheet for the "index" page. It includes a header section with a background color of #f0f0f0, a body section with a background color of #fff, and a body-light section with a background color of #f0f0f0. It also includes sections for "CONTENIDO INICIAL DA PÁGINA", "CONTENIDO DE PÁGINA", and "CONTENIDO DE PÁGINA".
- Top Bar:** Shows the Eclipse menu bar (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help) and the Apache Geode Java API - Apache Geode NetDev [X] title bar.

```
File Edit View Navigator Source Adapters Run Debug Profile Team Tools Window Help AgentServerGSP - Space for Hellfire (X) Search (Ctrl+F)
```

```
Project Explorer LogonController.java MainController.java TestController.java UploadImageController.java logon.jsp administrator.jsp exception.jsp
```

```
Source History
```

```
body {  
    padding: 15px;  
    border-radius: 10px;  
    background-color: #333333;  
    margin-bottom: 20px;  
}
```

```
body.light {  
    color: #333333;  
    background: #e6e6fa;  
    border-color: #ccc;  
}
```

```
h1 {  
    font-size: 2em;  
    font-weight: bold;  
    color: #333333;  
}
```

```
h2 {  
    font-size: 1.5em;  
    font-weight: bold;  
    color: #333333;  
}
```

```
h3 {  
    font-size: 1.2em;  
    font-weight: bold;  
    color: #333333;  
}
```

```
h4 {  
    font-size: 1.1em;  
    font-weight: bold;  
    color: #333333;  
}
```

```
h5 {  
    font-size: 1.0em;  
    font-weight: bold;  
    color: #333333;  
}
```

```
h6 {  
    font-size: 0.9em;  
    font-weight: bold;  
    color: #333333;  
}
```

```
p {  
    font-size: 1em;  
    font-weight: normal;  
    color: #333333;  
}
```

```
input {  
    width: 100%;  
    background-color: #f0f0f0;  
    color: #333333;  
    padding: 5px;  
    border-radius: 5px;  
    border: none;  
    margin-top: 20px;  
    font-weight: bold;  
}
```

```
.logon {  
    width: 100%;  
    height: 100%;  
    background-color: #f0f0f0;  
    color: #333333;  
    padding: 5px;  
    border-radius: 5px;  
    border: none;  
    margin-top: 20px;  
    font-weight: bold;  
}
```

```
.logon .text {  
    width: 100%;  
    height: 100%;  
    background-color: transparent;  
    color: #333333;  
    padding: 5px 15px;  
    border-radius: 5px;  
    border: none;  
    font-weight: bold;  
}
```

```
.logon .text input {  
    width: 100%;  
    height: 100%;  
    background-color: transparent;  
    color: #333333;  
    padding: 5px 15px;  
    border-radius: 5px;  
    border: none;  
}
```



The screenshot shows the Apache NetBeans IDE interface with the following details:

- Project Tree:** The project is named "AgenteServidorPS". It contains a "Web Pages" folder with files like "index.jsp", "index.htm", "index.html", "index.php", and "index.txt". There are also "Source Packages" and "Model" sections.
- Code Editor:** The main editor window displays Java code for a controller class, specifically the "LogoutController.java" file. The code includes annotations such as @WebServlet, @RequestWrapper, and @FacesConfig. It handles requests for URLs like "/LogoutController/logout.jsp", "/LogoutController/logout.htm", and "/LogoutController/logout.html". The code uses JSTL and EL expressions for navigation and session management.
- Toolbars and Menus:** Standard NetBeans menus like File, Edit, View, New, Source, AddFile, Run, Debug, Profile, Team, Tools, Window, Help are visible at the top. A toolbar with icons for file operations is located above the code editor.
- Status Bar:** The bottom status bar shows the path "src\LogoutController.java - Apache NetBeans [1,1]" and the word "Search Ctrl+F".

configuracoes.jsp



duvidas.jsp

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer (Left):** Shows the project structure for "AgentSevicioGU". It includes packages like "WebPages", "META-INF", "WEB-INF", "log", "util", "index", "controllers", "model", and "utils".
- Java Editor (Center):** Displays the code for "NuevoTipoDocumentoController.java". The code handles the creation of new document types via a POST request. It uses annotations like @Controller, @RequestMapping, and @ModelAttribute. It interacts with a service layer through a "NuevoTipoDocumentoService" interface.
- Top Bar:** Shows the standard Eclipse menu items (File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help) and the Apache Felix Gogo interface.
- Bottom Status Bar:** Shows the message "VALIDACIONES DE TIPO EXISTENTE".

```
package org.agentes.servicio.controller;

import org.agentes.servicio.model.NuevoTipoDocumento;
import org.agentes.servicio.service.NuevoTipoDocumentoService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.support.RedirectAttributes;

@Controller
@RequestMapping(value = "/nuevotipodocumento", method = RequestMethod.POST)
public class NuevoTipoDocumentoController {

    @Autowired
    private NuevoTipoDocumentoService servicio;

    @RequestMapping(value = "/guardar", method = RequestMethod.POST)
    public ModelAndView guardar(@ModelAttribute("nuevo") NuevoTipoDocumento nuevo,
                               RedirectAttributes redirectAttributes) {
        servicio.guardar(nuevo);
        redirectAttributes.addFlashAttribute("msj", "Guardado con exito");
        return new ModelAndView("redirect:/nuevotipodocumento");
    }
}
```



## editar-dados.jsp

```
<%@ page sessionType="request" language="JAVA" %>

<!-- Header: Dados do usuário
Session: sessão de usuário
Usário: prof. Filipe e Carla -->


<meta http-equiv="refresh" content="0; url=editardados.jsp?msg=usuário+salvo+com+sucesso!">
<script>
    window.location.href = "editardados.jsp?msg=usuário+salvo+com+sucesso!";
</script>

<%@ page sessionType="request" language="JAVA" %>

<%@ page sessionType="request" language="JAVA" %>
```

Screenshot of the Apache NetBeans IDE showing the 'editor-data.css' file in the 'Stylesheets' category of the 'AgentSevletGSP' project. The code defines a 'body' style with a background image and a 'background-color' of #f0f0f0. It also includes a 'form' style with a padding of 10px and a 'background-color' of #fff. A 'div' style is defined with a height of 100% and a 'background-color' of #e6e6fa. The 'body' style has a 'margin-top' of -10px.

```
body {
    background: url('img/bg.jpg');
    background-color: #f0f0f0;
}
form {
    padding: 10px;
    background-color: #fff;
}
div {
    height: 100%;
    background-color: #e6e6fa;
}
body {
    margin-top: -10px;
}
```

Screenshot of the Apache NetBeans IDE showing the 'editor-data.css' file in the 'Stylesheets' category of the 'AgentSevletGSP' project. The code defines a 'body' style with a background image and a 'background-color' of #f0f0f0. It also includes a 'form' style with a padding of 10px and a 'background-color' of #fff. A 'div' style is defined with a height of 100% and a 'background-color' of #e6e6fa. The 'body' style has a 'margin-top' of -10px.

```
body {
    background: url('img/bg.jpg');
    background-color: #f0f0f0;
}
form {
    padding: 10px;
    background-color: #fff;
}
div {
    height: 100%;
    background-color: #e6e6fa;
}
body {
    margin-top: -10px;
}
```

Screenshot of the Apache NetBeans IDE showing the 'editor-data.css' file in the 'Stylesheets' category of the 'AgentSevletGSP' project. The code defines a 'body' style with a background image and a 'background-color' of #f0f0f0. It also includes a 'form' style with a padding of 10px and a 'background-color' of #fff. A 'div' style is defined with a height of 100% and a 'background-color' of #e6e6fa. The 'body' style has a 'margin-top' of -10px.

```
body {
    background: url('img/bg.jpg');
    background-color: #f0f0f0;
}
form {
    padding: 10px;
    background-color: #fff;
}
div {
    height: 100%;
    background-color: #e6e6fa;
}
body {
    margin-top: -10px;
}
```

Project Explorer

```

Project: AgentDashboard
  -> Web Pages
    -> META-INF
    -> controller
    -> view
      -> home
        -> jsp
          -> index.jsp
          -> login.jsp
          -> register.jsp
          -> registerSuccess.jsp
          -> footer.jsp
          -> edit-profile.jsp
          -> kiosk.jsp
          -> help.jsp
          -> info.jsp
          -> success.jsp
          -> upload-file.jsp
        -> js
        -> stylesheets
        -> images
      -> web.xml
    -> config
    -> controller
    -> services
      -> AgentController.java
      -> CelularController.java
      -> CaptchaController.java
      -> CelularController.java
      -> EditorEditorController.java
      -> EmailSenderController.java
      -> HelpController.java
      -> LogoController.java
      -> MarcadorController.java
      -> MarcadorController.java
      -> UploadImageController.java
      -> UploadImageController.java
    -> model
      -> Celular
      -> Marcador
      -> Marcador
      -> Marcador
    -> util

```

Source

```

Index.jsp (18 lines)
  1 // Agente é nome de usuário
  2 function getCookie(name) {
  3   var cookieName = name + "=" + escape(document.cookie);
  4   var begin = cookieName.indexOf(";" + name + "=");
  5   if (begin == -1) begin = 0;
  6   var end = cookieName.indexOf(";", begin);
  7   if (end == -1) end = cookieName.length;
  8   return decodeURIComponent(cookieName.substring(begin + 1, end));
  9 }
 10
 11 // Aplicar o tema do site
 12 function applyTheme() {
 13   var tema = "verde";
 14   if (localStorage.getItem("tema") != null) {
 15     tema = localStorage.getItem("tema");
 16   }
 17   document.body.classList.toggle("light", tema === "light");
 18   updateLocalStorage();
 19 }
 20
 21 // Adiciona nome usado a sessão
 22 function topNome() {
 23   document.getElementById("nomeToggle").value = tema;
 24   tema = document.body.classList.contains("light") ? "light" : "dark";
 25   document.cookie = "tema=" + tema + "; path=/base-agents/";
 26   localStorage.setItem("tema", tema);
 27   updateLocalStorage();
 28 }
 29
 30 // Sinaliza se nome da sessão
 31 function updateLocalStorage() {
 32   var elementoId = "nomeToggle";
 33   var contexto = document.getElementById(elementoId).value;
 34   document.body.classList.contains("light") ? tema : tema;
 35 }
 36
 37 // Tema
 38 document.getElementById("themeToggle").addEventListener("click", toggleTheme);
 39
 40 // Instruções no topo ao carregar o página
 41 applyLocalStorage();
 42
 43 </script>
 44 </head>
 45 </html>

```

## home.jsp

Project Explorer

```

Project: AgentDashboard
  -> Web Pages
    -> META-INF
    -> controller
    -> view
      -> home
        -> jsp
          -> index.jsp
          -> login.jsp
          -> register.jsp
          -> registerSuccess.jsp
          -> footer.jsp
          -> edit-profile.jsp
          -> kiosk.jsp
          -> help.jsp
          -> info.jsp
          -> success.jsp
          -> upload-file.jsp
        -> js
        -> stylesheets
        -> images
      -> web.xml
    -> config
    -> controller
    -> services
      -> AgentController.java
      -> CelularController.java
      -> CaptchaController.java
      -> CelularController.java
      -> EditorEditorController.java
      -> EmailSenderController.java
      -> HelpController.java
      -> LogoController.java
      -> MarcadorController.java
      -> MarcadorController.java
      -> UploadImageController.java
      -> UploadImageController.java
    -> model
      -> Celular
      -> Marcador
      -> Marcador
      -> Marcador
    -> util

```

Source

```

index.jsp (18 lines)
  1 <%@ page contentType="text/html; charset=ISO-8859-1" %>
  2 <%
  3   /**
  4    * Pagina inicial da Plataforma
  5    * Desenvolvida para Agentes
  6    * Criado para Filiados e Clientes
  7    *
  8    * Tema é a página central do sistema. Aqui é utilizada visualização
  9    * de topo interativo, essa visualização preenche, preferencialmente
 10   * todos os elementos da interface com informações que podem ser
 11   * usadas para navegar entre as páginas do sistema.
 12   *
 13   * Tema é aplicado dinamicamente por localStorage,
 14   * garantindo consistência visual em todo o sistema.
 15   */
 16
 17 String crm = request.getParameter("crm");
 18 String cepInclusao = (String) session.getAttribute("cepInclusao");
 19
 20 // Cookies = nome usado na sessão global
 21 String nomeCookies = "Nome";
 22 javax.servlet.http.Cookie[] cookies = request.getCookies();
 23
 24 if (cookies != null) {
 25   for (javax.servlet.http.Cookie c : cookies) {
 26     if (c.getName().equals("Nome")) {
 27       temaCookies = c.getValue();
 28     }
 29   }
 30 }
 31
 32 <!--INCLUDES-->
 33 <!-- Header -->
 34 <link href="<%=request.getContextPath()%>/css/bootstrap.min.css" rel="stylesheet"/>
 35 <link href="<%=request.getContextPath()%>/css/bootstrap-theme.min.css" rel="stylesheet"/>
 36 <link href="<%=request.getContextPath()%>/css/bootstrap-theme.css.map" rel="stylesheet"/>
 37 <!-- Scripts -->
 38 <script src="<%=request.getContextPath()%>/js/jquery-3.1.1.min.js" type="text/javascript">
 39 <script src="<%=request.getContextPath()%>/js/bootstrap.min.js" type="text/javascript">
 40 <script src="<%=request.getContextPath()%>/js/bootstrap-theme.min.js" type="text/javascript">
 41 <script src="<%=request.getContextPath()%>/js/bootstrap-theme.css.map" type="text/javascript">
 42
 43 <script>

```

```
<%@page language="java" %>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
        <title>Index Page</title>
    </head>
    <body>
        <a href="index.html">Index Page</a>
    </body>
</html>
```

```
/* Header Starts (External) */

header {
    width: 100px;
    height: 100px;
    background-color: #edf2ff;
    border: 1px solid #ccc;
    border-radius: 10px;
    position: fixed;
    top: 0;
    left: 0;
    padding: 2px;
    border-right: 1px solid var(--accent);
    -webkit-overflow-scrolling: auto;
    overflow-y: auto;
    transition: background .4s ease, color .4s ease;
}

body {
    margin: 0;
}

.map {
    position: absolute;
    left: 100px;
    top: 0;
    width: calc(100% - 100px);
    height: 100px;
}

/* Header Ends (External)

.list-item {
    width: 100px;
    background-color: #edf2ff;
    border: 1px solid #ccc;
    border-radius: 10px;
    margin: 10px 0;
    padding: 10px;
    font-size: 14px;
    color: #333;
}
```

The screenshot shows the AgentServer IDE interface with the following details:

- Project Tree:** Shows the project structure under "Projects".
- Code Editor:** Displays Java code for a controller class named `HomeController.java`. The code includes annotations like `@Controller`, `@RequestMapping`, and `@GetMapping`. It defines several methods including `index`, `home`, `about`, `contact`, `configuration`, `developer`, `privacy`, `termsAndConditions`, and `version`.
- Toolbars:** Standard Java development toolbars for file operations, navigation, and search.
- Status Bar:** Shows the path "AgentServer[PU] - Home Page (File 1)" and the date "2023-01-10 10:45:21".

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
```

```
Project: AgentGearsGP
```

```
Source History Search (Ctrl+F)
```

```
1 // CARGA STADOS SALVOS DE LOCALSTORAGE
2
3 function cargarStados() {
4     const saved = JSON.parse(localStorage.getItem(STATES_KEY)) || "{}";
5
6     saved.forEach((item, index) => {
7         let shape;
8
9         if (item.type === "rectangle") {
10             shape = new google.maps.Rectangle();
11             shape.setBounds(
12                 new google.maps.LatLng(item.lat, item.lng),
13                 new google.maps.LatLng(item.lat + item.height,
14                     item.lng + item.width));
15             shape.setOptions({
16                 strokeColor: "#000000",
17                 strokeDasharray: [4, 4],
18                 strokeWeight: 1,
19                 fillColor: "#E6E6FA",
20                 fillColorOpacity: 0.15,
21                 fillOpacity: 0.15,
22                 map: map
23             });
24
25             map.addOverlay(shape);
26
27         } else if (item.type === "circle") {
28             shape = new google.maps.Circle();
29             shape.setCenter({lat: item.lat, lng: item.lng});
30             shape.setRadius(item.radius);
31             shape.setOptions({
32                 strokeColor: "#000000",
33                 strokeDasharray: [4, 4],
34                 strokeWeight: 1,
35                 fillColor: "#E6E6FA",
36                 fillOpacity: 0.15,
37                 map: map
38             });
39
40             map.addOverlay(shape);
41
42         }
43
44         shapesDatabase.push(shape, index);
45     });
46
47     google.maps.event.addListener(shape, "click", (e) => {
48         infoShape.setOptions({
49             content: "Estado: " + e.latLng.lat() + ", " + e.latLng.lng(),
50             position: e.latLng,
51             align: "center",
52             closeBoxMargin: "5px 5px 5px 5px",
53             closeBoxURL: "close.png",
54             disableAutoPan: true,
55             pixelOffset: new google.Point(0, -10),
56             title: "Estado"
57         });
58
59         infoShape.open(map);
60     });
61
62     infoShape.setOptions({
63         content: "Estado: " + saved.states[0].lat + ", " + saved.states[0].lng,
64         position: saved.states[0],
65         align: "center",
66         closeBoxMargin: "5px 5px 5px 5px",
67         closeBoxURL: "close.png",
68         disableAutoPan: true,
69         pixelOffset: new google.Point(0, -10),
70         title: "Estado"
71     });
72
73     infoShape.open(map);
74
75     infoShape.setOptions({
76         content: "Cargando...",
77         position: map.getCenter(),
78         align: "center",
79         closeBoxMargin: "5px 5px 5px 5px",
80         closeBoxURL: "close.png",
81         disableAutoPan: true,
82         pixelOffset: new google.Point(0, -10),
83         title: "Cargando..."
84     });
85
86     infoShape.open(map);
87
88     infoShape.setOptions({
89         content: "Cargando...",
90         position: map.getCenter(),
91         align: "center",
92         closeBoxMargin: "5px 5px 5px 5px",
93         closeBoxURL: "close.png",
94         disableAutoPan: true,
95         pixelOffset: new google.Point(0, -10),
96         title: "Cargando..."
97     });
98
99     infoShape.open(map);
100
101     infoShape.setOptions({
102         content: "Cargando...",
103         position: map.getCenter(),
104         align: "center",
105         closeBoxMargin: "5px 5px 5px 5px",
106         closeBoxURL: "close.png",
107         disableAutoPan: true,
108         pixelOffset: new google.Point(0, -10),
109         title: "Cargando..."
110     });
111
112     infoShape.open(map);
113
114     infoShape.setOptions({
115         content: "Cargando...",
116         position: map.getCenter(),
117         align: "center",
118         closeBoxMargin: "5px 5px 5px 5px",
119         closeBoxURL: "close.png",
120         disableAutoPan: true,
121         pixelOffset: new google.Point(0, -10),
122         title: "Cargando..."
123     });
124
125     infoShape.open(map);
126
127     infoShape.setOptions({
128         content: "Cargando...",
129         position: map.getCenter(),
130         align: "center",
131         closeBoxMargin: "5px 5px 5px 5px",
132         closeBoxURL: "close.png",
133         disableAutoPan: true,
134         pixelOffset: new google.Point(0, -10),
135         title: "Cargando..."
136     });
137
138     infoShape.open(map);
139
140     infoShape.setOptions({
141         content: "Cargando...",
142         position: map.getCenter(),
143         align: "center",
144         closeBoxMargin: "5px 5px 5px 5px",
145         closeBoxURL: "close.png",
146         disableAutoPan: true,
147         pixelOffset: new google.Point(0, -10),
148         title: "Cargando..."
149     });
150
151     infoShape.open(map);
152
153     infoShape.setOptions({
154         content: "Cargando...",
155         position: map.getCenter(),
156         align: "center",
157         closeBoxMargin: "5px 5px 5px 5px",
158         closeBoxURL: "close.png",
159         disableAutoPan: true,
160         pixelOffset: new google.Point(0, -10),
161         title: "Cargando..."
162     });
163
164     infoShape.open(map);
165
166     infoShape.setOptions({
167         content: "Cargando...",
168         position: map.getCenter(),
169         align: "center",
170         closeBoxMargin: "5px 5px 5px 5px",
171         closeBoxURL: "close.png",
172         disableAutoPan: true,
173         pixelOffset: new google.Point(0, -10),
174         title: "Cargando..."
175     });
176
177     infoShape.open(map);
178
179     infoShape.setOptions({
180         content: "Cargando...",
181         position: map.getCenter(),
182         align: "center",
183         closeBoxMargin: "5px 5px 5px 5px",
184         closeBoxURL: "close.png",
185         disableAutoPan: true,
186         pixelOffset: new google.Point(0, -10),
187         title: "Cargando..."
188     });
189
190     infoShape.open(map);
191
192     infoShape.setOptions({
193         content: "Cargando...",
194         position: map.getCenter(),
195         align: "center",
196         closeBoxMargin: "5px 5px 5px 5px",
197         closeBoxURL: "close.png",
198         disableAutoPan: true,
199         pixelOffset: new google.Point(0, -10),
200         title: "Cargando..."
201     });
202
203     infoShape.open(map);
204
205     infoShape.setOptions({
206         content: "Cargando...",
207         position: map.getCenter(),
208         align: "center",
209         closeBoxMargin: "5px 5px 5px 5px",
210         closeBoxURL: "close.png",
211         disableAutoPan: true,
212         pixelOffset: new google.Point(0, -10),
213         title: "Cargando..."
214     });
215
216     infoShape.open(map);
217
218     infoShape.setOptions({
219         content: "Cargando...",
220         position: map.getCenter(),
221         align: "center",
222         closeBoxMargin: "5px 5px 5px 5px",
223         closeBoxURL: "close.png",
224         disableAutoPan: true,
225         pixelOffset: new google.Point(0, -10),
226         title: "Cargando..."
227     });
228
229     infoShape.open(map);
230
231     infoShape.setOptions({
232         content: "Cargando...",
233         position: map.getCenter(),
234         align: "center",
235         closeBoxMargin: "5px 5px 5px 5px",
236         closeBoxURL: "close.png",
237         disableAutoPan: true,
238         pixelOffset: new google.Point(0, -10),
239         title: "Cargando..."
240     });
241
242     infoShape.open(map);
243
244     infoShape.setOptions({
245         content: "Cargando...",
246         position: map.getCenter(),
247         align: "center",
248         closeBoxMargin: "5px 5px 5px 5px",
249         closeBoxURL: "close.png",
250         disableAutoPan: true,
251         pixelOffset: new google.Point(0, -10),
252         title: "Cargando..."
253     });
254
255     infoShape.open(map);
256
257     infoShape.setOptions({
258         content: "Cargando...",
259         position: map.getCenter(),
260         align: "center",
261         closeBoxMargin: "5px 5px 5px 5px",
262         closeBoxURL: "close.png",
263         disableAutoPan: true,
264         pixelOffset: new google.Point(0, -10),
265         title: "Cargando..."
266     });
267
268     infoShape.open(map);
269
270     infoShape.setOptions({
271         content: "Cargando...",
272         position: map.getCenter(),
273         align: "center",
274         closeBoxMargin: "5px 5px 5px 5px",
275         closeBoxURL: "close.png",
276         disableAutoPan: true,
277         pixelOffset: new google.Point(0, -10),
278         title: "Cargando..."
279     });
280
281     infoShape.open(map);
282
283     infoShape.setOptions({
284         content: "Cargando...",
285         position: map.getCenter(),
286         align: "center",
287         closeBoxMargin: "5px 5px 5px 5px",
288         closeBoxURL: "close.png",
289         disableAutoPan: true,
290         pixelOffset: new google.Point(0, -10),
291         title: "Cargando..."
292     });
293
294     infoShape.open(map);
295
296     infoShape.setOptions({
297         content: "Cargando...",
298         position: map.getCenter(),
299         align: "center",
300         closeBoxMargin: "5px 5px 5px 5px",
301         closeBoxURL: "close.png",
302         disableAutoPan: true,
303         pixelOffset: new google.Point(0, -10),
304         title: "Cargando..."
305     });
306
307     infoShape.open(map);
308
309     infoShape.setOptions({
310         content: "Cargando...",
311         position: map.getCenter(),
312         align: "center",
313         closeBoxMargin: "5px 5px 5px 5px",
314         closeBoxURL: "close.png",
315         disableAutoPan: true,
316         pixelOffset: new google.Point(0, -10),
317         title: "Cargando..."
318     });
319
320     infoShape.open(map);
321
322     infoShape.setOptions({
323         content: "Cargando...",
324         position: map.getCenter(),
325         align: "center",
326         closeBoxMargin: "5px 5px 5px 5px",
327         closeBoxURL: "close.png",
328         disableAutoPan: true,
329         pixelOffset: new google.Point(0, -10),
330         title: "Cargando..."
331     });
332
333     infoShape.open(map);
334
335     infoShape.setOptions({
336         content: "Cargando...",
337         position: map.getCenter(),
338         align: "center",
339         closeBoxMargin: "5px 5px 5px 5px",
340         closeBoxURL: "close.png",
341         disableAutoPan: true,
342         pixelOffset: new google.Point(0, -10),
343         title: "Cargando..."
344     });
345
346     infoShape.open(map);
347
348     infoShape.setOptions({
349         content: "Cargando...",
350         position: map.getCenter(),
351         align: "center",
352         closeBoxMargin: "5px 5px 5px 5px",
353         closeBoxURL: "close.png",
354         disableAutoPan: true,
355         pixelOffset: new google.Point(0, -10),
356         title: "Cargando..."
357     });
358
359     infoShape.open(map);
360
361     infoShape.setOptions({
362         content: "Cargando...",
363         position: map.getCenter(),
364         align: "center",
365         closeBoxMargin: "5px 5px 5px 5px",
366         closeBoxURL: "close.png",
367         disableAutoPan: true,
368         pixelOffset: new google.Point(0, -10),
369         title: "Cargando..."
370     });
371
372     infoShape.open(map);
373
374     infoShape.setOptions({
375         content: "Cargando...",
376         position: map.getCenter(),
377         align: "center",
378         closeBoxMargin: "5px 5px 5px 5px",
379         closeBoxURL: "close.png",
380         disableAutoPan: true,
381         pixelOffset: new google.Point(0, -10),
382         title: "Cargando..."
383     });
384
385     infoShape.open(map);
386
387     infoShape.setOptions({
388         content: "Cargando...",
389         position: map.getCenter(),
390         align: "center",
391         closeBoxMargin: "5px 5px 5px 5px",
392         closeBoxURL: "close.png",
393         disableAutoPan: true,
394         pixelOffset: new google.Point(0, -10),
395         title: "Cargando..."
396     });
397
398     infoShape.open(map);
399
400     infoShape.setOptions({
401         content: "Cargando...",
402         position: map.getCenter(),
403         align: "center",
404         closeBoxMargin: "5px 5px 5px 5px",
405         closeBoxURL: "close.png",
406         disableAutoPan: true,
407         pixelOffset: new google.Point(0, -10),
408         title: "Cargando..."
409     });
410
411     infoShape.open(map);
412
413     infoShape.setOptions({
414         content: "Cargando...",
415         position: map.getCenter(),
416         align: "center",
417         closeBoxMargin: "5px 5px 5px 5px",
418         closeBoxURL: "close.png",
419         disableAutoPan: true,
420         pixelOffset: new google.Point(0, -10),
421         title: "Cargando..."
422     });
423
424     infoShape.open(map);
425
426     infoShape.setOptions({
427         content: "Cargando...",
428         position: map.getCenter(),
429         align: "center",
430         closeBoxMargin: "5px 5px 5px 5px",
431         closeBoxURL: "close.png",
432         disableAutoPan: true,
433         pixelOffset: new google.Point(0, -10),
434         title: "Cargando..."
435     });
436
437     infoShape.open(map);
438
439     infoShape.setOptions({
440         content: "Cargando...",
441         position: map.getCenter(),
442         align: "center",
443         closeBoxMargin: "5px 5px 5px 5px",
444         closeBoxURL: "close.png",
445         disableAutoPan: true,
446         pixelOffset: new google.Point(0, -10),
447         title: "Cargando..."
448     });
449
450     infoShape.open(map);
451
452     infoShape.setOptions({
453         content: "Cargando...",
454         position: map.getCenter(),
455         align: "center",
456         closeBoxMargin: "5px 5px 5px 5px",
457         closeBoxURL: "close.png",
458         disableAutoPan: true,
459         pixelOffset: new google.Point(0, -10),
460         title: "Cargando..."
461     });
462
463     infoShape.open(map);
464
465     infoShape.setOptions({
466         content: "Cargando...",
467         position: map.getCenter(),
468         align: "center",
469         closeBoxMargin: "5px 5px 5px 5px",
470         closeBoxURL: "close.png",
471         disableAutoPan: true,
472         pixelOffset: new google.Point(0, -10),
473         title: "Cargando..."
474     });
475
476     infoShape.open(map);
477
478     infoShape.setOptions({
479         content: "Cargando...",
480         position: map.getCenter(),
481         align: "center",
482         closeBoxMargin: "5px 5px 5px 5px",
483         closeBoxURL: "close.png",
484         disableAutoPan: true,
485         pixelOffset: new google.Point(0, -10),
486         title: "Cargando..."
487     });
488
489     infoShape.open(map);
490
491     infoShape.setOptions({
492         content: "Cargando...",
493         position: map.getCenter(),
494         align: "center",
495         closeBoxMargin: "5px 5px 5px 5px",
496         closeBoxURL: "close.png",
497         disableAutoPan: true,
498         pixelOffset: new google.Point(0, -10),
499         title: "Cargando..."
500     });
501
502     infoShape.open(map);
503
504     infoShape.setOptions({
505         content: "Cargando...",
506         position: map.getCenter(),
507         align: "center",
508         closeBoxMargin: "5px 5px 5px 5px",
509         closeBoxURL: "close.png",
510         disableAutoPan: true,
511         pixelOffset: new google.Point(0, -10),
512         title: "Cargando..."
513     });
514
515     infoShape.open(map);
516
517     infoShape.setOptions({
518         content: "Cargando...",
519         position: map.getCenter(),
520         align: "center",
521         closeBoxMargin: "5px 5px 5px 5px",
522         closeBoxURL: "close.png",
523         disableAutoPan: true,
524         pixelOffset: new google.Point(0, -10),
525         title: "Cargando..."
526     });
527
528     infoShape.open(map);
529
530     infoShape.setOptions({
531         content: "Cargando...",
532         position: map.getCenter(),
533         align: "center",
534         closeBoxMargin: "5px 5px 5px 5px",
535         closeBoxURL: "close.png",
536         disableAutoPan: true,
537         pixelOffset: new google.Point(0, -10),
538         title: "Cargando..."
539     });
540
541     infoShape.open(map);
542
543     infoShape.setOptions({
544         content: "Cargando...",
545         position: map.getCenter(),
546         align: "center",
547         closeBoxMargin: "5px 5px 5px 5px",
548         closeBoxURL: "close.png",
549         disableAutoPan: true,
550         pixelOffset: new google.Point(0, -10),
551         title: "Cargando..."
552     });
553
554     infoShape.open(map);
555
556     infoShape.setOptions({
557         content: "Cargando...",
558         position: map.getCenter(),
559         align: "center",
560         closeBoxMargin: "5px 5px 5px 5px",
561         closeBoxURL: "close.png",
562         disableAutoPan: true,
563         pixelOffset: new google.Point(0, -10),
564         title: "Cargando..."
565     });
566
567     infoShape.open(map);
568
569     infoShape.setOptions({
570         content: "Cargando...",
571         position: map.getCenter(),
572         align: "center",
573         closeBoxMargin: "5px 5px 5px 5px",
574         closeBoxURL: "close.png",
575         disableAutoPan: true,
576         pixelOffset: new google.Point(0, -10),
577         title: "Cargando..."
578     });
579
580     infoShape.open(map);
581
582     infoShape.setOptions({
583         content: "Cargando...",
584         position: map.getCenter(),
585         align: "center",
586         closeBoxMargin: "5px 5px 5px 5px",
587         closeBoxURL: "close.png",
588         disableAutoPan: true,
589         pixelOffset: new google.Point(0, -10),
590         title: "Cargando..."
591     });
592
593     infoShape.open(map);
594
595     infoShape.setOptions({
596         content: "Cargando...",
597         position: map.getCenter(),
598         align: "center",
599         closeBoxMargin: "5px 5px 5px 5px",
600         closeBoxURL: "close.png",
601         disableAutoPan: true,
602         pixelOffset: new google.Point(0, -10),
603         title: "Cargando..."
604     });
605
606     infoShape.open(map);
607
608     infoShape.setOptions({
609         content: "Cargando...",
610         position: map.getCenter(),
611         align: "center",
612         closeBoxMargin: "5px 5px 5px 5px",
613         closeBoxURL: "close.png",
614         disableAutoPan: true,
615         pixelOffset: new google.Point(0, -10),
616         title: "Cargando..."
617     });
618
619     infoShape.open(map);
620
621     infoShape.setOptions({
622         content: "Cargando...",
623         position: map.getCenter(),
624         align: "center",
625         closeBoxMargin: "5px 5px 5px 5px",
626         closeBoxURL: "close.png",
627         disableAutoPan: true,
628         pixelOffset: new google.Point(0, -10),
629         title: "Cargando..."
630     });
631
632     infoShape.open(map);
633
634     infoShape.setOptions({
635         content: "Cargando...",
636         position: map.getCenter(),
637         align: "center",
638         closeBoxMargin: "5px 5px 5px 5px",
639         closeBoxURL: "close.png",
640         disableAutoPan: true,
641         pixelOffset: new google.Point(0, -10),
642         title: "Cargando..."
643     });
644
645     infoShape.open(map);
646
647     infoShape.setOptions({
648         content: "Cargando...",
649         position: map.getCenter(),
650         align: "center",
651         closeBoxMargin: "5px 5px 5px 5px",
652         closeBoxURL: "close.png",
653         disableAutoPan: true,
654         pixelOffset: new google.Point(0, -10),
655         title: "Cargando..."
656     });
657
658     infoShape.open(map);
659
660     infoShape.setOptions({
661         content: "Cargando...",
662         position: map.getCenter(),
663         align: "center",
664         closeBoxMargin: "5px 5px 5px 5px",
665         closeBoxURL: "close.png",
666         disableAutoPan: true,
667         pixelOffset: new google.Point(0, -10),
668         title: "Cargando..."
669     });
670
671     infoShape.open(map);
672
673     infoShape.setOptions({
674         content: "Cargando...",
675         position: map.getCenter(),
676         align: "center",
677         closeBoxMargin: "5px 5px 5px 5px",
678         closeBoxURL: "close.png",
679         disableAutoPan: true,
680         pixelOffset: new google.Point(0, -10),
681         title: "Cargando..."
682     });
683
684     infoShape.open(map);
685
686     infoShape.setOptions({
687         content: "Cargando...",
688         position: map.getCenter(),
689         align: "center",
690         closeBoxMargin: "5px 5px 5px 5px",
691         closeBoxURL: "close.png",
692         disableAutoPan: true,
693         pixelOffset: new google.Point(0, -10),
694         title: "Cargando..."
695     });
696
697     infoShape.open(map);
698
699     infoShape.setOptions({
700         content: "Cargando...",
701         position: map.getCenter(),
702         align: "center",
703         closeBoxMargin: "5px 5px 5px 5px",
704         closeBoxURL: "close.png",
705         disableAutoPan: true,
706         pixelOffset: new google.Point(0, -10),
707         title: "Cargando..."
708     });
709
710     infoShape.open(map);
711
712     infoShape.setOptions({
713         content: "Cargando...",
714         position: map.getCenter(),
715         align: "center",
716         closeBoxMargin: "5px 5px 5px 5px",
717         closeBoxURL: "close.png",
718         disableAutoPan: true,
719         pixelOffset: new google.Point(0, -10),
720         title: "Cargando..."
721     });
722
723     infoShape.open(map);
724
725     infoShape.setOptions({
726         content: "Cargando...",
727         position: map.getCenter(),
728         align: "center",
729         closeBoxMargin: "5px 5px 5px 5px",
730         closeBoxURL: "close.png",
731         disableAutoPan: true,
732         pixelOffset: new google.Point(0, -10),
733         title: "Cargando..."
734     });
735
736     infoShape.open(map);
737
738     infoShape.setOptions({
739         content: "Cargando...",
740         position: map.getCenter(),
741         align: "center",
742         closeBoxMargin: "5px 5px 5px 5px",
743         closeBoxURL: "close.png",
744         disableAutoPan: true,
745         pixelOffset: new google.Point(0, -10),
746         title: "Cargando..."
747     });
748
749     infoShape.open(map);
750
751     infoShape.setOptions({
752         content: "Cargando...",
753         position: map.getCenter(),
754         align: "center",
755         closeBoxMargin: "5px 5px 5px 5px",
756         closeBoxURL: "close.png",
757         disableAutoPan: true,
758         pixelOffset: new google.Point(0, -10),
759         title: "Cargando..."
760     });
761
762     infoShape.open(map);
763
764     infoShape.setOptions({
765         content: "Cargando...",
766         position: map.getCenter(),
767         align: "center",
768         closeBoxMargin: "5px 5px 5px 5px",
769         closeBoxURL: "close.png",
770         disableAutoPan: true,
771         pixelOffset: new google.Point(0, -10),
772         title: "Cargando..."
773     });
774
775     infoShape.open(map);
776
777     infoShape.setOptions({
778         content: "Cargando...",
779         position: map.getCenter(),
780         align: "center",
781         closeBoxMargin: "5px 5px 5px 5px",
782         closeBoxURL: "close.png",
783         disableAutoPan: true,
784         pixelOffset: new google.Point(0, -10),
785         title: "Cargando..."
786     });
787
788     infoShape.open(map);
789
790     infoShape.setOptions({
791         content: "Cargando...",
792         position: map.getCenter(),
793         align: "center",
794         closeBoxMargin: "5px 5px 5px 5px",
795         closeBoxURL: "close.png",
796         disableAutoPan: true,
797         pixelOffset: new google.Point(0, -10),
798         title: "Cargando..."
799     });
799
800     infoShape.open(map);
801
802     infoShape.setOptions({
803         content: "Cargando...",
804         position: map.getCenter(),
805         align: "center",
806         closeBoxMargin: "5px 5px 5px 5px",
807         closeBoxURL: "close.png",
808         disableAutoPan: true,
809         pixelOffset: new google.Point(0, -10),
810         title: "Cargando..."
811     });
811
812     infoShape.open(map);
813
814     infoShape.setOptions({
815         content: "Cargando...",
816         position: map.getCenter(),
817         align: "center",
818         closeBoxMargin: "5px 5px 5px 5px",
819         closeBoxURL: "close.png",
820         disableAutoPan: true,
821         pixelOffset: new google.Point(0, -10),
822         title: "Cargando..."
823     });
823
824     infoShape.open(map);
825
826     infoShape.setOptions({
827         content: "Cargando...",
828         position: map.getCenter(),
829         align: "center",
830         closeBoxMargin: "5px 5px 5px 5px",
831         closeBoxURL: "close.png",
832         disableAutoPan: true,
833         pixelOffset: new google.Point(0, -10),
834         title: "Cargando..."
835     });
835
836     infoShape.open(map);
837
837     infoShape.setOptions({
838         content: "Cargando...",
839         position: map.getCenter(),
840         align: "center",
841         closeBoxMargin: "5px 5px 5px 5px",
842         closeBoxURL: "close.png",
843         disableAutoPan: true,
844         pixelOffset: new google.Point(0, -10),
845         title: "Cargando..."
846     });
846
847     infoShape.open(map);
848
848     infoShape.setOptions({
849         content: "Cargando...",
850         position: map.getCenter(),
851         align: "center",
852         closeBoxMargin: "5px 5px 5px 5px",
853         closeBoxURL: "close.png",
854         disableAutoPan: true,
855         pixelOffset: new google.Point(0, -10),
856         title: "Cargando..."
857     });
857
858     infoShape.open(map);
859
859     infoShape.setOptions({
860         content: "Cargando...",
861         position: map.getCenter(),
862         align: "center",
863         closeBoxMargin: "5px 5px 5px 5px",
864         closeBoxURL: "close.png",
865         disableAutoPan: true,
866         pixelOffset: new google.Point(0, -10),
867         title: "Cargando..."
868     });
868
869     infoShape.open(map);
870
870     infoShape.setOptions({
871         content: "Cargando...",
872         position: map.getCenter(),
873         align: "center",
874         closeBoxMargin: "5px 5px 5px 5px",
875         closeBoxURL: "close.png",
876         disableAutoPan: true,
877         pixelOffset: new google.Point(0, -10),
878         title: "Cargando..."
879     });
879
880     infoShape.open(map);
881
881     infoShape.setOptions({
882         content: "Cargando...",
883         position: map.getCenter(),
884         align: "center",
885         closeBoxMargin: "5px 5px 5px 5px",
886         closeBoxURL: "close.png",
887         disableAutoPan: true,
888         pixelOffset: new google.Point(0, -10),
889         title: "Cargando..."
890     });
890
891     infoShape.open(map);
892
892     infoShape.setOptions({
893         content: "Cargando...",
894         position: map.getCenter(),
895         align: "center",
896         closeBoxMargin: "5px 5px 5px 5px",
897         closeBoxURL: "close.png",
898         disableAutoPan: true,
899         pixelOffset: new google.Point(0, -10),
900         title: "Cargando..."
901     });
901
902     infoShape.open(map);
903
903     infoShape.setOptions({
904         content: "Cargando...",
905         position: map.getCenter(),
906         align: "center",
907         closeBoxMargin: "5px 5px 5px 5px",
908         closeBoxURL: "close.png",
909         disableAutoPan: true,
910         pixelOffset: new google.Point(0, -10),
911         title: "Cargando..."
912     });
912
913     infoShape.open(map);
914
914     infoShape.setOptions({
915         content: "Cargando...",
916         position: map.getCenter(),
917         align: "center",
918         closeBoxMargin: "5px 5px 5px 5px",
919         closeBoxURL: "close.png",
920         disableAutoPan: true,
921         pixelOffset: new google.Point(0, -10),
922         title: "Cargando..."
923     });
923
924     infoShape.open(map);
925
925     infoShape.setOptions({
926         content: "Cargando...",
927         position: map.getCenter(),
928         align: "center",
929         closeBoxMargin: "5px 5px 5px 5px",
930         closeBoxURL: "close.png",
931         disableAutoPan: true,
932         pixelOffset: new google.Point(0, -10),
933         title: "Cargando..."
934     });
934
935     infoShape.open(map);
936
936     infoShape.setOptions({
937         content: "Cargando...",
938         position: map.getCenter(),
939         align: "center",
940         closeBoxMargin: "5px 5px 5px 5px",
941         closeBoxURL: "close.png",
942         disableAutoPan: true,
943         pixelOffset: new google.Point(0, -10),
944         title: "Cargando..."
945     });
945
946     infoShape.open(map);
947
947     infoShape.setOptions({
948         content: "Cargando...",
949         position: map.getCenter(),
950         align: "center",
951         closeBoxMargin: "5px 5px 5px 5px",
952         closeBoxURL: "close.png",
953         disableAutoPan: true,
954         pixelOffset: new google.Point(0, -10),
955         title: "Cargando..."
956     });
956
957     infoShape.open(map);
958
958     infoShape.setOptions({
959         content: "Cargando...",
960         position: map.getCenter(),
961         align: "center",
962         closeBoxMargin: "5px 5px 5px 5px",
963         closeBoxURL: "close.png",
964         disableAutoPan: true,
965         pixelOffset: new google.Point(0, -10),
966         title: "Cargando..."
967     });
967
968     infoShape.open(map);
969
969     infoShape.setOptions({
970         content: "Cargando...",
971         position: map.getCenter(),
972         align: "center",
973         closeBoxMargin: "5px 5px 5px 5px",
974         closeBoxURL: "close.png",
975         disableAutoPan: true,
976         pixelOffset: new google.Point(0, -10),
977         title: "Cargando..."
978     });
978
979     infoShape.open(map);
980
980     infoShape.setOptions({
981         content: "Cargando...",
982         position: map.getCenter(),
983         align: "center",
984         closeBoxMargin: "5px 5px 5px 5px",
985         closeBoxURL: "close.png",
986         disableAutoPan: true,
987         pixelOffset: new google.Point(0, -10),
988         title: "Cargando..."
989     });
989
990     infoShape.open(map);
991
991     infoShape.setOptions({
992         content: "Cargando...",
993         position: map.getCenter(),
994         align: "center",
995         closeBoxMargin: "5px 5px 5px 5px",
996         closeBoxURL: "close.png",
997         disableAutoPan: true,
998         pixelOffset: new google.Point(0, -10),
999         title: "Cargando..."
1000    });
1000
```

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
```

```
Project: AgentGearsGP
```

```
Source History Search (Ctrl+F)
```

```
1 // CARGA DE ESTADOS GUERREROS
2
3 function cargarGuerreros() {
4     const saved = JSON.parse(localStorage.getItem(GUERREROS_KEY)) || "{}";
5
6     saved.forEach((item, index) => {
7         let guerrero;
8
9         if (item.type === "rectangle") {
10             guerrero = new google.maps.Rectangle();
11             guerrero.setBounds(
12                 new google.maps.LatLng(item.lat, item.lng),
13                 new google.maps.LatLng(item.lat + item.height,
14                     item.lng + item.width));
15             guerrero.setOptions({
16                 strokeColor: "#000000",
17                 strokeDasharray: [4, 4],
18                 strokeWeight: 1,
19                 fillColor: "#E6E6FA",
20                 fillColorOpacity: 0.15,
21                 fillOpacity: 0.15,
22                 map: map
23             });
24
25             map.addOverlay(guerrero);
26
27         } else if (item.type === "circle") {
28             guerrero = new google.maps.Circle();
29             guerrero.setCenter({lat: item.lat, lng: item.lng});
30             guerrero.setRadius(item.radius);
31             guerrero.setOptions({
32                 strokeColor: "#000000",
33                 strokeDasharray: [4, 4],
34                 strokeWeight: 1,
35                 fillColor: "#E6E6FA",
36                 fillOpacity: 0.15,
37                 map: map
38             });
39
40             map.addOverlay(guerrero);
41
42         }
43
44         guerrerosDatabase.push(guerrero, index);
45     });
46
47     infoGuerrero.setOptions({
48         content: "Guerrero: " + guerrerosDatabase[0].lat + ", " + guerrerosDatabase[0].lng,
49         position: guerrerosDatabase[0],
50         align: "center",
51         closeBoxMargin: "5px 5px 5px 5px",
52         closeBoxURL: "close.png",
53         disableAutoPan: true,
54         pixelOffset: new google.Point(0, -10),
55         title: "Guerrero"
56     });
57
58     infoGuerrero.open(map);
59
60     infoGuerrero.setOptions({
61         content: "Cargando...",
62         position: map.getCenter(),
63         align: "center",
64         closeBoxMargin: "5px
```

Project: AgentGearsGP

```
Source History Search (Ctrl+F) [File] [Edit] [View] [Navigate] [Source] [Refactor] [Run] [Debug] [Profile] [Team] [Tools] [Window] [Help]
```

```
function salvarNota(market, let, img) {
    if (!market.idMercador) return;

    if (img) {
        market.addListener("change", () => salvarNotaPoligono(market, let.leting.lng(), let.leting.lat()));
    }

    market.addListener("change", () => marketInfo(market));
    market.push(market);
}

/*
 * Salvar nota quando clica no marcador
 */
function salvarNota(market) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    market.leting.setCaption(nota);
    market.leting.setAddress(escapeHTML(market.let));
}

/*
 * Salvar nota quando o usuário clica no mapa
 */
function salvarNota(mapa) {
    if (!confirm("Deseja salvar essa posição?")) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}

/*
 * Salvar nota quando o usuário clica no mapa e adiciona
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}

/*
 * Salvar nota quando o usuário clica no mapa e adiciona
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}
```

Project: AgentGearsGP

```
Source History Search (Ctrl+F) [File] [Edit] [View] [Navigate] [Source] [Refactor] [Run] [Debug] [Profile] [Team] [Tools] [Window] [Help]
```

```
function salvarNota(market) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    market.leting.setCaption(nota);
    market.leting.setAddress(escapeHTML(market.let));
}

/*
 * Salvar nota quando o usuário clica no mapa
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}

/*
 * Salvar nota quando o usuário clica no mapa e adiciona
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}

/*
 * Salvar nota quando o usuário clica no mapa e adiciona
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}
```

Project: AgentGearsGP

```
Source History Search (Ctrl+F) [File] [Edit] [View] [Navigate] [Source] [Refactor] [Run] [Debug] [Profile] [Team] [Tools] [Window] [Help]
```

```
function salvarNota(market, let, img) {
    if (!market.idMercador) return;

    if (img) {
        market.addListener("change", () => salvarNotaPoligono(market, let.leting.lng(), let.leting.lat()));
    }

    market.addListener("change", () => marketInfo(market));
    market.push(market);
}

/*
 * Salvar nota quando clica no marcador
 */
function salvarNota(market) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    market.leting.setCaption(nota);
    market.leting.setAddress(escapeHTML(market.let));
}

/*
 * Salvar nota quando o usuário clica no mapa
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}

/*
 * Salvar nota quando o usuário clica no mapa e adiciona
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}

/*
 * Salvar nota quando o usuário clica no mapa e adiciona
 */
function salvarNota(mapa) {
    const nota = prompt("Márcador anotado", market.let);
    if (!nota) return;

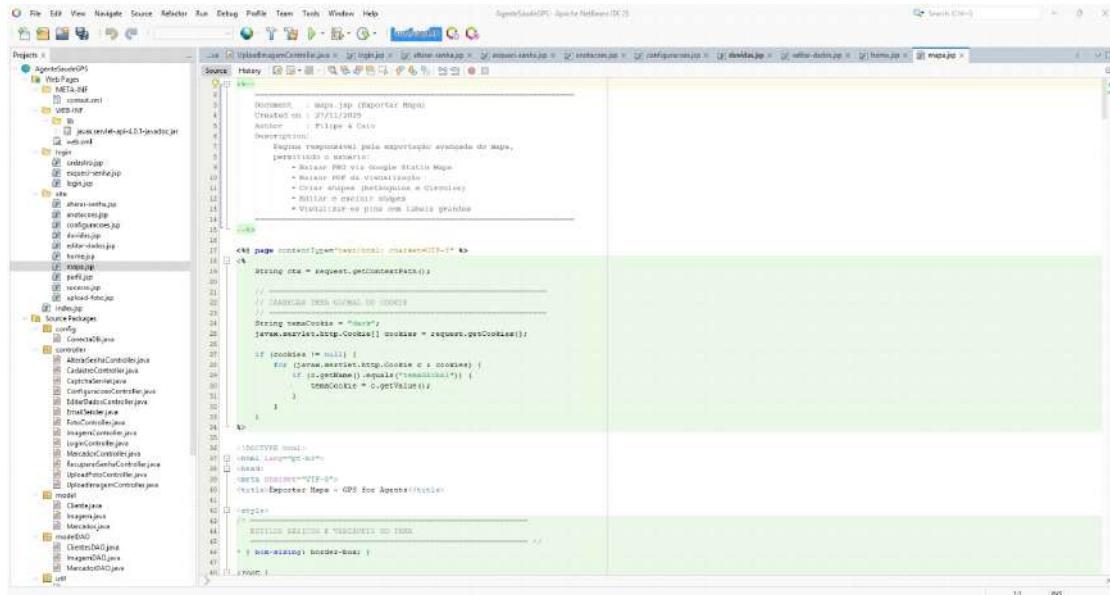
    const BASE = "http://localhost:8080";
    const nota = mapa.getMarker();
    const headers = {"Content-Type": "application/x-www-form-urlencoded"};
    const url = "http://www.mapasgeograficos.com.br/api/marcador/nova/" + nota.idMercador;
    const data = "latitude=" + nota.leting.lat() + "&longitude=" + nota.leting.lng();

    const options = {
        method: "POST",
        headers: headers,
        body: urlencoded(data)
    };

    fetch(BASE + url, options)
        .then(response => response.json())
        .then(data => {
            const marker = new google.maps.Marker({
                position: {lat: data.latitude, lng: data.longitude}
            });
            mapa.setMarker(marker);
        })
        .catch(error => console.error(error));
}
```

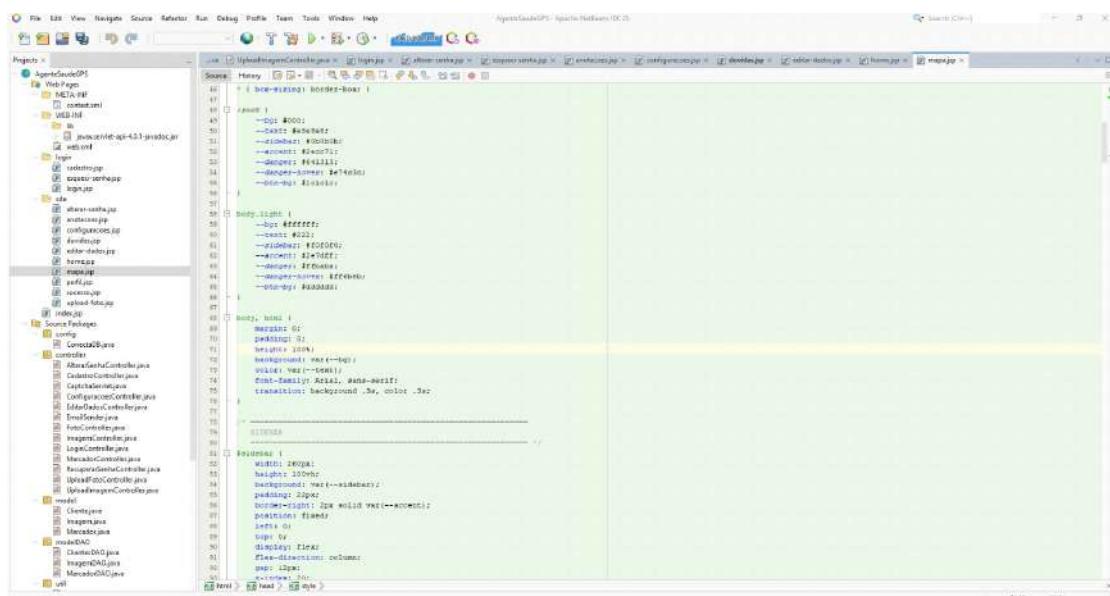


## mapa.jsp



Document : mapa.jsp (Reportar Bug)  
Created on : 27/11/2009  
Author : Felipe & Cauê  
Description:  
Esse componente põe exportação avançada do mapa,  
permittendo o usuário:  
• Salvar PDF via Google Static Map  
• Salvar em formato de Imagem  
• Criar mapas participantes e filiais  
• Baixar o PDF das imagens  
• Visualizar as primeiras labels grandes

```
<%@ page contentType="text/html; charset=UTF-8" %>
<% String cta = request.getContextPath(); %>
<%-- CÓDIGO PARA OBTER OS COOKIES --%>
String nomeCookie = "param";
java.util.List<Http.Cookie> cookies = request.getCookies();
if (cookies != null) {
    for (Http.Cookie cookie : cookies) {
        if (cookie.getName().equals("nomeCookie")) {
            nomeCookie = cookie.getValue();
        }
    }
}
<%-- EXECUTA JS PARA MUDAR O NOME DO COOKIES --%>
<%-- CÓDIGO PARA MUDAR O NOME DO TÍTULO --%>
<%-- CÓDIGO PARA MUDAR O NOME DA URL --%>
<%-- CÓDIGO PARA MUDAR O NOME DA URL --%>
```



```
body, html {
    margin: 0;
    padding: 0;
    height: 100%;
    background-color: #f0f0f0;
    color: #000000;
    font-family: Arial, sans-serif;
    transition: background .5s, color .3s;
}
#mapa {
    width: 100%;
    height: 100px;
    border: 1px solid #ccc;
    border-radius: 10px;
    position: relative;
    z-index: 1;
}
#mapa::before {
    content: '';
    width: 100px;
    height: 100px;
    border: 1px solid #ccc;
    border-radius: 10px;
    position: absolute;
    top: 0;
    left: 0;
    z-index: -1;
}
```

The screenshot shows the Apache NetBeans IDE interface with the following details:

- File Menu:** File, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Project Tree:** Shows the project structure with packages like `Algoritmos`, `Entidades`, `Modelos`, `Views`, and `utils`. Under `Views`, there are several JSP files: `index.jsp`, `indexAdmin.jsp`, `indexUser.jsp`, `login.jsp`, `menu.jsp`, `menuAdmin.jsp`, `menuUser.jsp`, `reservas.jsp`, `reservasAdmin.jsp`, `reservasUser.jsp`, `salir.jsp`, `seguimiento.jsp`, `seguimientoAdmin.jsp`, `seguimientoUser.jsp`, `verReserva.jsp`, and `verReservaAdmin.jsp`.
- Code Editor:** The main window displays the `index.jsp` file content. The code includes CSS styles for background colors and fonts, and Java code for session management and database connections. It also contains JSTL tags for displaying user information and a welcome message.







## perfil.jsp

```
<%@ page contentType="text/html; charset=UTF-8" %>
<% String cta = request.getAttribute("idMensageiro");
Integer idMensageiro = (Integer) session.getAttribute("idMensageiro");
if (idMensageiro == null) {
    response.sendRedirect(cta + "login.jsp");
}
String nome = (String) session.getAttribute("nomeMensageiro");
String email = (String) session.getAttribute("emailMensageiro");
String cep = (String) session.getAttribute("cepMensageiro");
// Toma global via cookie
String temPerfil = "false";
java.util.List<Cookie> cookies = request.getCookies();
if (cookies != null) {
    for (Cookie c : cookies) {
        if (c.getName().equals("temPerfil")) {
            if (c.getValue().equals("trueGlobal")) temPerfil = c.getValue();
        }
    }
}
//ENCRIPTA CTIA
String Langage="pt-BR";
ctia
//envia
//envia com o HTTP
//url:New_Perfil - URL for Agents@x11a
//link href="https://node_x11a.xmpp.brasil.chat/buddycard.html?user=rodrigobr"
//link href="https://node_x11a.xmpp.brasil.chat/buddycard.html?user=rodrigobr&nick=rodrigobr";
//criar
%>
```

```
<%@ page contentType="text/html; charset=UTF-8" %>
<% String cta = request.getAttribute("idMensageiro");
Integer idMensageiro = (Integer) session.getAttribute("idMensageiro");
if (idMensageiro == null) {
    response.sendRedirect(cta + "login.jsp");
}
String nome = (String) session.getAttribute("nomeMensageiro");
String email = (String) session.getAttribute("emailMensageiro");
String cep = (String) session.getAttribute("cepMensageiro");
// Toma global via cookie
String temPerfil = "false";
java.util.List<Cookie> cookies = request.getCookies();
if (cookies != null) {
    for (Cookie c : cookies) {
        if (c.getName().equals("temPerfil")) {
            if (c.getValue().equals("trueGlobal")) temPerfil = c.getValue();
        }
    }
}
//ENCRIPTA CTIA
String Langage="pt-BR";
ctia
//envia
//envia com o HTTP
//url:New_Perfil - URL for Agents@x11a
//link href="https://node_x11a.xmpp.brasil.chat/buddycard.html?user=rodrigobr"
//link href="https://node_x11a.xmpp.brasil.chat/buddycard.html?user=rodrigobr&nick=rodrigobr";
//criar
%>
```



sucesso.jsp

The screenshot shows the Apache NetBeans IDE interface. The top menu bar includes File, Edit, View, Navigate, Source, Adapter, Run, Debug, Profile, Team, Tools, Window, Help, and a search bar. The left sidebar displays the project structure under 'Project 1' with several Java files (e.g., AgentServerGUIServlet.java, Main.java, MainDAO.java, MainController.java) and a CSS file (style.css). The main editor area shows the content of style.css, which contains CSS rules for various HTML elements like body, h1, and .logos. The code editor has syntax highlighting and a vertical scrollbar.

```
body {  
background-color: #e0e0e0;  
color: #000000;  
font-family: Arial, Helvetica, sans-serif;  
margin: 0;  
padding: 0;  
display: flex;  
align-items: center;  
justify-content: center;  
background: linear-gradient(45deg, #e0e0e0, #d0d0d0);  
}  
  
h1 {  
background-color: #e0e0e0;  
color: #000000;  
font-family: Arial, Helvetica, sans-serif;  
margin: 0;  
padding: 0;  
display: flex;  
align-items: center;  
justify-content: center;  
background: linear-gradient(45deg, #e0e0e0, #d0d0d0);  
}  
  
/* Container */  
  
.container {  
background-color: #fff;  
border-radius: 10px;  
width: 300px;  
border: 1px solid #ccc;  
text-align: center;  
border-bottom: 2px solid #ccc; /* Gradient */  
background-color: #f0f0f0; /* Gradient */  
}  
  
/* Buttons */  
  
.button {  
background-color: #007bff;  
border: none;  
border-radius: 5px;  
color: white;  
font-size: 16px;  
padding: 10px;  
width: 100%;  
border-radius: 10px;  
text-align: center;  
border: 1px solid #007bff;  
background-color: #007bff; /* Gradient */  
}
```

## upload-foto.jsp

```
Project: AgentServerGP
Source Packages: AgentServerGP
  - AgentServerGP
    - WebPages
      - WEB-INF
        - controllers
        - WEB-INF
          - accesscontrol-api-0.0.1.jar
        - web.xml
      - login
        - redirect.jsp
        - register.jsp
        - login.jsp
      - site
        - about.jsp
        - contact.jsp
        - configuration.jsp
        - favIcon.jsp
        - help.jsp
        - home.jsp
        - map.jsp
        - pdf.jsp
        - search.jsp
        - static.html
    - index.jsp
  - Source Packages
  - ControllerBase
  - com.agent
    - controllers
      - AboutController.java
      - CaptchaController.java
      - ConfigurationController.java
      - HomeController.java
      - LogoutController.java
      - LoginController.java
      - PhotoController.java
      - ImageController.java
      - LogoutController.java
      - MarcadoController.java
      - ResponseSelectionController.java
      - UserProfileController.java
    - model
      - Clientes
      - Marcados
      - Marcado
    - modelDAO
      - ClientesDAO.java
      - MarcadosDAO.java
      - MarcadoDAO.java
    - util
Source: ProfileController.java
1 package com.agent.controllers;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.RequestMapping;
5 import org.springframework.web.bind.annotation.RequestMethod;
6 import org.springframework.web.bind.annotation.RequestParam;
7 import org.springframework.web.bind.annotation.SessionAttributes;
8 import org.springframework.web.bind.support.SessionStatus;
9 import org.springframework.web.servlet.ModelAndView;
10
11 import com.agent.model.Marcado;
12 import com.agent.model.Clientes;
13 import com.agent.modelDAO.ClientesDAO;
14 import com.agent.modelDAO.MarcadosDAO;
15 import com.agent.modelDAO.MarcadoDAO;
16
17 import java.util.List;
18
19 /**
20 * Controller para gerenciar os perfis de usuário
21 */
22 @Controller
23 @RequestMapping(value = "/profile")
24 @SessionAttributes("user")
25 public class ProfileController {
26
27     /**
28     * Salva ou altera o perfil do usuário
29     * 
30     * @param marcado
31     * @param nome
32     * @param email
33     * @param telefone
34     * @param endereço
35     * @param cidade
36     * @param bairro
37     * @param cep
38     * @param numero
39     * @param complemento
40     * @param tipo
41     * @param status
42     * @param id
43     * @param session
44     * @param response
45     * @param status
46     */
47     @RequestMapping(value = "edit", method = RequestMethod.POST)
48     public ModelAndView salvar(@ModelAttribute("marcado") Marcado marcado,
49                               @RequestParam("nome") String nome,
50                               @RequestParam("email") String email,
51                               @RequestParam("telefone") String telefone,
52                               @RequestParam("endereco") String endereco,
53                               @RequestParam("cidade") String cidade,
54                               @RequestParam("bairro") String bairro,
55                               @RequestParam("cep") String cep,
56                               @RequestParam("numero") String numero,
57                               @RequestParam("complemento") String complemento,
58                               @RequestParam("tipo") String tipo,
59                               @RequestParam("status") String status,
60                               @RequestParam("id") Integer id,
61                               HttpSession session,
62                               HttpServletResponse response,
63                               SessionStatus status) {
64
65         if (id != null) {
66             marcado.setId(id);
67             marcado.setNome(nome);
68             marcado.setEmail(email);
69             marcado.setTelefone(telefone);
70             marcado.setEndereco(endereco);
71             marcado.setCidade(cidade);
72             marcado.setBairro(bairro);
73             marcado.setCep(cep);
74             marcado.setNumero(numero);
75             marcado.setComplemento(complemento);
76             marcado.setTipo(tipo);
77             marcado.setStatus(status);
78
79             UserServices userServices = new UserServices();
80             Clientes cliente = userServices.getCliente(session);
81             marcado.setCliente(cliente);
82             marcado.setUsuario(userServices.getUsuario(session));
83             marcado.setAtivo(true);
84             marcado.setCriadoEm(new Date());
85             marcado.setAtualizadoEm(new Date());
86
87             ClientesDAO clientesDAO = new ClientesDAO();
88             clientesDAO.salvar(marcado);
89
90             status.setComplete();
91             response.sendRedirect("profile");
92         } else {
93             marcado.setNome(nome);
94             marcado.setEmail(email);
95             marcado.setTelefone(telefone);
96             marcado.setEndereco(endereco);
97             marcado.setCidade(cidade);
98             marcado.setBairro(bairro);
99             marcado.setCep(cep);
100            marcado.setNumero(numero);
101            marcado.setComplemento(complemento);
102            marcado.setTipo(tipo);
103            marcado.setStatus(status);
104
105            UserServices userServices = new UserServices();
106            Clientes cliente = userServices.getCliente(session);
107            marcado.setCliente(cliente);
108            marcado.setUsuario(userServices.getUsuario(session));
109            marcado.setAtivo(true);
110            marcado.setCriadoEm(new Date());
111            marcado.setAtualizadoEm(new Date());
112
113            ClientesDAO clientesDAO = new ClientesDAO();
114            clientesDAO.salvar(marcado);
115
116            status.setComplete();
117            response.sendRedirect("profile");
118        }
119
120        return new ModelAndView("redirect:/profile");
121    }
122
123    /**
124     * Remove um perfil de usuário
125     * 
126     * @param id
127     * @param session
128     * @param response
129     */
130    @RequestMapping(value = "remove", method = RequestMethod.GET)
131    public ModelAndView remove(@RequestParam("id") Integer id,
132                               HttpSession session,
133                               HttpServletResponse response) {
134
135        UserServices userServices = new UserServices();
136        Clientes cliente = userServices.getCliente(session);
137        Marcado marcado = cliente.getMarcado(id);
138
139        marcado.setAtivo(false);
140
141        ClientesDAO clientesDAO = new ClientesDAO();
142        clientesDAO.remover(marcado);
143
144        session.setAttribute("user", null);
145
146        response.sendRedirect("login");
147
148        return new ModelAndView("redirect:/login");
149    }
150
151    /**
152     * Redireciona para a página de login
153     * 
154     * @param session
155     * @param response
156     */
157    @RequestMapping(value = "login", method = RequestMethod.GET)
158    public ModelAndView login(HttpSession session,
159                             HttpServletResponse response) {
160
161        String nomeCookie = "user";
162        UserServices userServices = new UserServices();
163        ClientesDAO clientesDAO = new ClientesDAO();
164
165        Clientes cliente = userServices.getCliente(session);
166
167        if (cliente == null) {
168            response.sendRedirect("login");
169        } else {
170            response.sendRedirect("profile");
171        }
172
173        return new ModelAndView("redirect:/login");
174    }
175
176    /**
177     * Redireciona para a página de login
178     * 
179     * @param session
180     * @param response
181     */
182    @RequestMapping(value = "logout", method = RequestMethod.GET)
183    public ModelAndView logout(HttpSession session,
184                              HttpServletResponse response) {
185
186        session.setAttribute("user", null);
187
188        response.sendRedirect("login");
189
190        return new ModelAndView("redirect:/login");
191    }
192}
```

Screenshot of the Eclipse IDE showing the code editor for a JSP file named 'upload-data.jsp'. The code is a CSS file embedded within the JSP. It defines a class 'anexo' with various styles including width, padding, background-color, border, and font-size.

```
<%@ page language="java" %>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Upload Data</title>
    <link href="css/upload-data.css" type="text/css" rel="stylesheet"/>
</head>
<body>
    <div id="content">
        <div id="form">
            <form action="upload-data" method="post" enctype="multipart/form-data">
                <input type="file" name="file" style="width: 100%; height: 100%; padding: 10px; margin-bottom: 10px; background-color: #f0f0f0; border: none; border-radius: 10px; outline: none; font-size: 14px; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);>
                <input type="button" value="Upload" style="width: 100px; height: 40px; background-color: #337ab7; color: white; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);>
            </form>
        </div>
    </div>
</body>
```

Screenshot of the Eclipse IDE showing the code editor for a JSP file named 'index.jsp'. The code includes a CSS block for a 'anexo' class with styles like position: fixed, top: 0px, right: 0px, background: transparent, border: none, font-size: 10px, color: var(--text-color), and font-weight: normal. It also contains Java code for handling file uploads and displaying user information.

```
<%@ page language="java" %>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Index</title>
    <link href="css/index.css" type="text/css" rel="stylesheet"/>
</head>
<body>
    <div id="content">
        <div id="header">
            <img alt="Logo" style="width: 100px; height: 100px; border-radius: 50%; border: 2px solid #337ab7; margin-bottom: 10px;"/>
            <div>
                <h1>Welcome, <span><%= request.getAttribute("username") %></span>!</h1>
                <p>Your profile:<br/>
                    <img alt="Profile Picture" style="width: 100px; height: 100px; border-radius: 50%; border: 2px solid #337ab7; margin-bottom: 10px;"/>
                    <span><%= request.getAttribute("profilePicture") %></span>
                </p>
                <form action="update-profile" method="post" style="margin-top: 20px; border: 1px solid #337ab7; padding: 10px; border-radius: 10px; background-color: #f0f0f0; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);>
                    <input type="text" name="username" value="<%= request.getAttribute("username") %>" style="width: 100%; height: 40px; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);"/>
                    <input type="text" name="email" value="<%= request.getAttribute("email") %>" style="width: 100%; height: 40px; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);"/>
                    <input type="text" name="password" value="<%= request.getAttribute("password") %>" style="width: 100%; height: 40px; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);"/>
                    <input type="text" name="newPassword" style="width: 100%; height: 40px; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);"/>
                    <input type="text" name="confirmPassword" style="width: 100%; height: 40px; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);"/>
                    <input type="submit" value="Update Profile" style="width: 100%; height: 40px; background-color: #337ab7; color: white; border: none; border-radius: 5px; font-size: 14px; border: 1px solid #337ab7; transition: all 0.2s; opacity: 0.8; transform: translate3d(0, 0, 0);"/>
                </form>
            </div>
        </div>
    </div>
</body>
```

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

AgentGeotabGPI - Apache NetBeans [0:25]

Project AgentGeotabGPI

Source History

```

1<?xml version="1.0" encoding="UTF-8"?>
2<!-- Generated by Apache NetBeans -->
3<form action="#" method="post">
4    <input type="file" name="file" accept="image/*">
5    <input type="button" value="Upload" onclick="document.getElementById('file').click();"/>
6</form>
7
```

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

AgentGeotabGPI - Apache NetBeans [0:75]

Project AgentGeotabGPI

Source History

```

1<?xml version="1.0" encoding="UTF-8"?>
2<!-- Generated by Apache NetBeans -->
3<script>
4    function toggleTheme() {
5        var theme = getCookie("themeGlobal");
6        if (localStorage.getItem("theme") == "dark") {
7            theme = "light";
8        } else {
9            theme = "dark";
10        }
11        document.body.classList.toggle("light");
12        document.body.classList.toggle("dark");
13        localStorage.setItem("theme", theme);
14        updateThemeIcon();
15    }
16
17    function updateThemeIcon() {
18        var theme = getCookie("themeGlobal");
19        if (theme == "light") {
20            document.body.classList.toggle("light");
21            document.cookie = "themeGlobal=" + theme + "; path=/; max-age=150000";
22            localStorage.setItem("theme", theme);
23            updateThemeIcon();
24        }
25    }
26
27    function getCookie(name) {
28        var cookieValue = document.cookie.match(new RegExp('(^| )' + name + '=([^;]+)'));
29        if (cookieValue) {
30            return cookieValue[2];
31        }
32        return null;
33    }
34
35    function applyTheme() {
36        var theme = getCookie("themeGlobal");
37        if (theme == "light") {
38            document.body.classList.toggle("light");
39            document.body.classList.toggle("dark");
40            localStorage.setItem("theme", theme);
41        }
42    }
43
44    document.addEventListener("DOMContentLoaded", () => {
45        applyTheme();
46    });
47
48    document.getElementById("themeToggle").addEventListener("click", () => {
49        toggleTheme();
50    });
51
52    document.getElementById("themePreview").addEventListener("change", () => {
53        previewImage = document.getElementById("previewImage");
54        previewImage.src = URL.createObjectURL(document.getElementById("themePreview").files[0]);
55    });
56
57    document.getElementById("themeInput").addEventListener("change", () => {
58        var fileInput = document.getElementById("themeInput");
59        var fileReader = new FileReader();
60        fileReader.onload = e => previewImage.src = URL.createObjectURL(e.target.result);
61        fileReader.readAsDataURL(fileInput.files[0]);
62    });
63
64    </script>
65
66</body>
67</html>
68
```

web.xml

The image displays two side-by-side screenshots of the Eclipse IDE interface, both titled "AgentForGPS - Apache NetBeans (8.7)".

**Left Window:**

- Project View:** Shows the project structure under "AgentForGPS". It includes a "Web Pages" folder containing "index.jsp", "index.html", "index.htm", "index.php", and "index.txt". A "WEB-INF" folder contains "web.xml", "lib", and "classes".
- Content View:** Displays the content of "web.xml". The XML code defines several servlets and their mappings:

```

<web-app version="2.5" encoding="UTF-8">
    <servlet>
        <servlet-name>LoginController</servlet-name>
        <servlet-class>controller.LoginController</servlet-class>
        <init-param>
            <param-name>loginPage</param-name>
            <param-value>/index.jsp</param-value>
        </init-param>
        <load-on-startup>1</load-on-startup>
    </servlet>
    <servlet-mapping>
        <servlet-name>LoginController</servlet-name>
        <url-pattern>/LoginController</url-pattern>
    </servlet-mapping>
    <servlet>
        <servlet-name>CadastroController</servlet-name>
        <servlet-class>controller.CadastroController</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>CadastroController</servlet-name>
        <url-pattern>/CadastroController</url-pattern>
    </servlet-mapping>
    <servlet>
        <servlet-name>DepoimentoService</servlet-name>
        <servlet-class>controller.DepoimentoService</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>DepoimentoService</servlet-name>
        <url-pattern>/DepoimentoService</url-pattern>
    </servlet-mapping>
    <servlet>
        <servlet-name>DepoimentoController</servlet-name>
        <servlet-class>controller.DepoimentoController</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>DepoimentoController</servlet-name>
        <url-pattern>/DepoimentoController</url-pattern>
    </servlet-mapping>
    <servlet>
        <servlet-name>BemPassearSenhaController</servlet-name>
        <servlet-class>controller.BemPassearSenhaController</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>BemPassearSenhaController</servlet-name>
        <url-pattern>/BemPassearSenhaController</url-pattern>
    </servlet-mapping>

```

**Right Window:**

- Project View:** Shows the same project structure as the left window.
- Content View:** Displays the content of "web.xml". The XML code is identical to the one in the left window, defining the same servlets and their mappings.

## 7. Repositório GitHub

Link do repositório do projeto: <https://github.com/Filipe-Lourenco7/AgentForGPS>

## 8. Referências Bibliográficas

DEITEL, Paul; DEITEL, Harvey. Java: Como Programar. 10. ed. São Paulo: Pearson, 2016.

Google. Google Maps API Documentation. Disponível em:

<https://developers.google.com/maps>

Oracle. Java EE Documentation. Disponível em: <https://www.oracle.com/java/>