

Java Advanced

21/08/2025

Prof. Dr. Marcel Stefan Wagner

Aula 06 – Spring Security e Oauth2

FIAP

Conceitos de *Spring Security*

Conceitos

Spring Security

- *Spring Security*

Quando adiciona-se a dependência do *Spring Security* no Projeto, por exemplo, utilizando-se o *Spring Initializr*, automaticamente todas as rotas são protegidas por padrão.

O *Spring Security* configura por padrão um usuário e uma senha.

Além disso, o *Spring Security* cria uma interface de *login* e de *logout*.

Implementando a Segurança Básica

The screenshot shows the Spring Initializr web application interface. The left side is highlighted with a red box and contains the following configuration:

- Project**: Maven (selected)
- Language**: Java (selected)
- Spring Boot**: 3.3.3 (selected)
- Project Metadata**:
 - Group: br.com.fiap
 - Artifact: SecurityBasic
 - Name: SecurityBasic
 - Description: Project for Spring Security
 - Package name: br.com.fiap.SecurityBasic
 - Packaging: Jar (selected)
- Java**: 21 (selected)

The right side is also highlighted with a red box and contains the following dependency sections:

- Dependencies**: ADD DEPENDENCIES... CTRL + B
- Spring Web** (WEB): Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.
- Spring Boot DevTools** (DEVELOPER TOOLS): Provides fast application restarts, LiveReload, and configurations for enhanced development experience.
- Spring Security** (SECURITY): Highly customizable authentication and access-control framework for Spring applications.

At the bottom, there are three buttons:

- GENERATE CTRL + ↵
- EXPLORE CTRL + SPACE
- SHARE...

The screenshot shows a Java application named `SecurityBasicApplication` running in an IDE. The code in `HttpController.java` defines two endpoints: a public route that returns a general message and a private route that returns a message specifically for authorized users.

```
package br.com.fiap.SecurityBasic.Controller;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HttpController {

    @GetMapping("/public")
    String publicRoute() {
        return "<h1>Public route, feel free to look around! </h1>";
    }

    @GetMapping("/private")
    String privateRoute() {
        return "<h1>Private route, only authorized by Tranquilo! </h1>";
    }
}
```

The IDE interface includes a project tree on the left, a toolbar at the bottom with various icons, and status bars at the top and bottom indicating file paths, build status, and system information.

The screenshot shows an IDE interface with the following details:

- File Menu:** File, Edit, View, Navigate, Code, Refactor, Build, Run, Tools, VCS, Window, Help.
- Project Explorer:** Shows the project structure under "SecurityBasic".
- Code Editors:** Three tabs are visible: `SecurityBasicApplication.java`, `HttpController.java` (selected), and `generated-requests.http`.

```
package br.com.fiap.SecurityBasic.Controller;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HttpController {

    @GetMapping("/public")
    String publicRoute() {
        return "<h1>Public route, feel free to look around! </h1>";
    }

    @GetMapping("/private")
    String privateRoute() {
        return "<h1>Private route, you need to be logged in to access it! </h1>";
    }
}
```

- Run Tab:** Shows the application `SecurityBasicApplication` is running.
- Console Tab:** Displays the application's logs. A red box highlights the line: `2024-08-30T00:39:45.514-03:00 INFO 3060 --- [SecurityBasic] b.c.f.S.SecurityBasicApplication : Started SecurityBasicApplication`.

```
2024-08-30T00:39:45.450-03:00 INFO 3060 --- [SecurityBasic] r$InitializeUserDetailsManagerConfigurer : Global AuthenticationManager configured with 'rememberMe' functionality
2024-08-30T00:39:45.485-03:00 INFO 3060 --- [SecurityBasic] n.s.b.d.o.OptionalLiveReloadServer : LiveReload server is running on port 35729
2024-08-30T00:39:45.507-03:00 INFO 3060 --- [SecurityBasic] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http://localhost:8080)
2024-08-30T00:39:45.514-03:00 INFO 3060 --- [SecurityBasic] b.c.f.S.SecurityBasicApplication : Started SecurityBasicApplication
2024-08-30T00:39:45.517-03:00 INFO 3060 --- [SecurityBasic] c.ConditionEvaluationDeltaLoggingListener : condition evaluation unchanged
```

- Bottom Bar:** Shows the file path `SecurityBasic > src > main > java > br > com > fiap > SecurityBasic > Controller > HttpController > publicRoute`, file encoding `UTF-8`, and other system information like date and time.



Public route, feel free to look around!

Deveria ser assim o retorno com o *endpoint public*

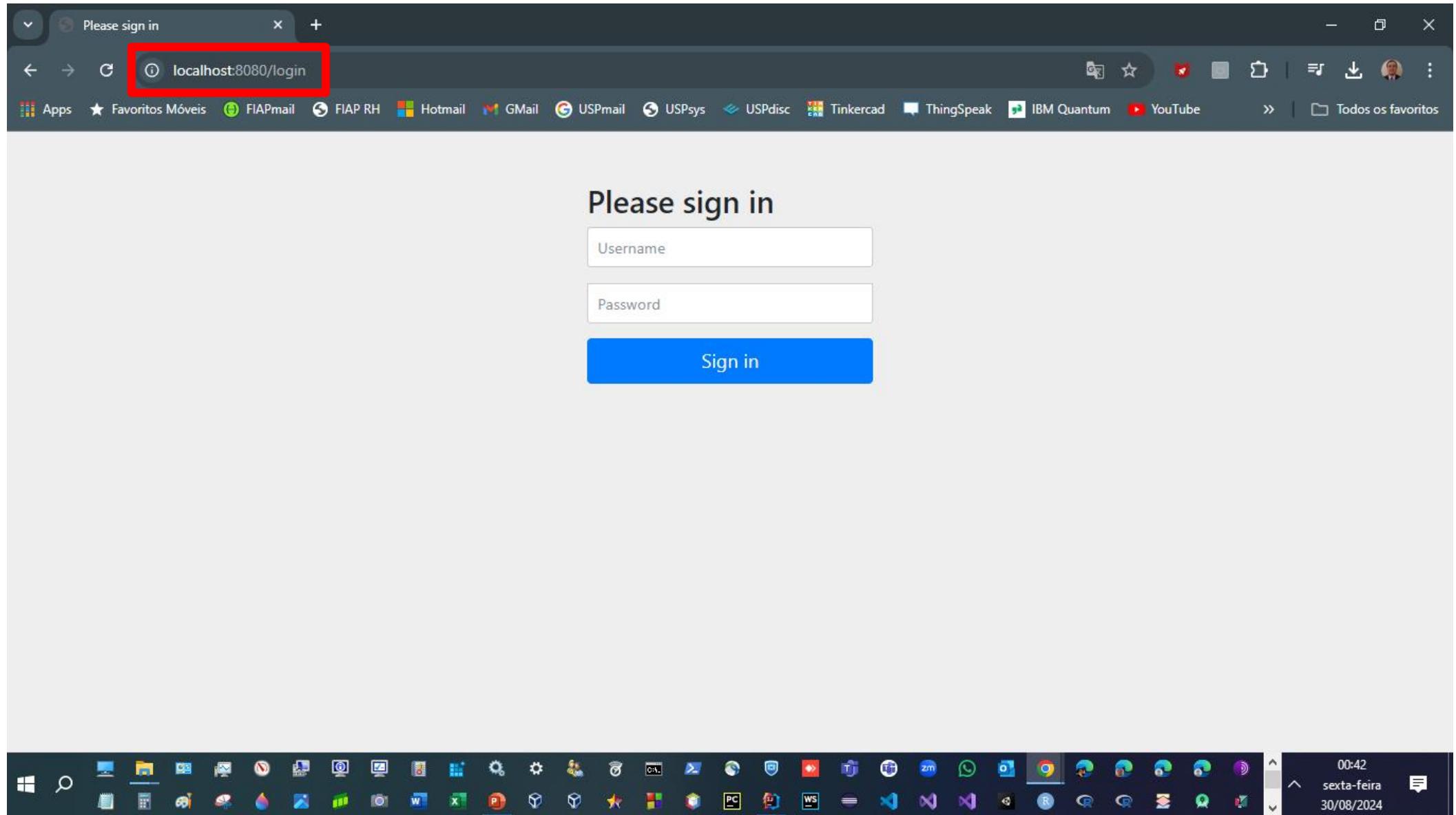


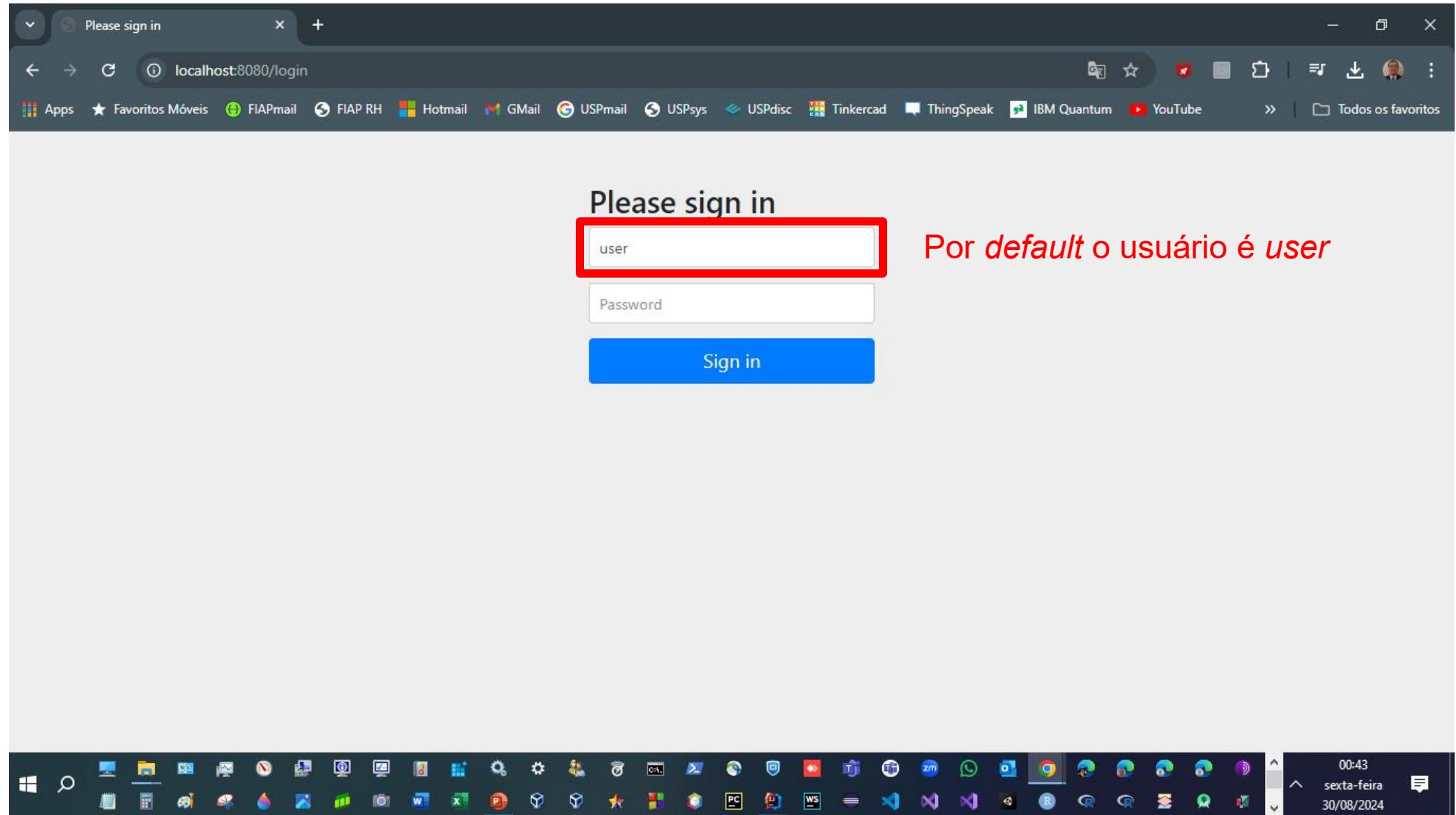


Private route, only authorized by Tranquilo!

Deveria ser assim o retorno com o *endpoint private*





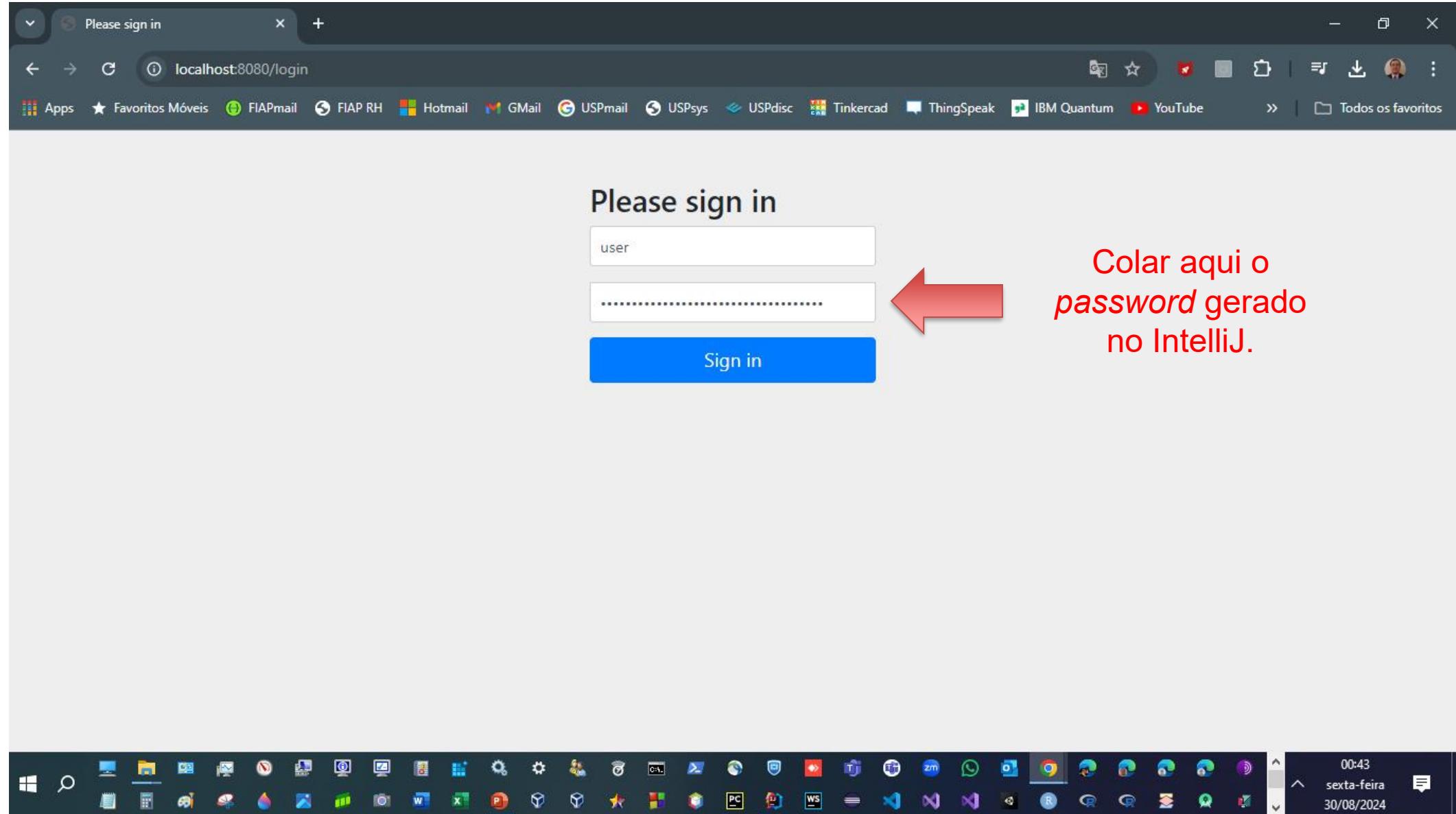


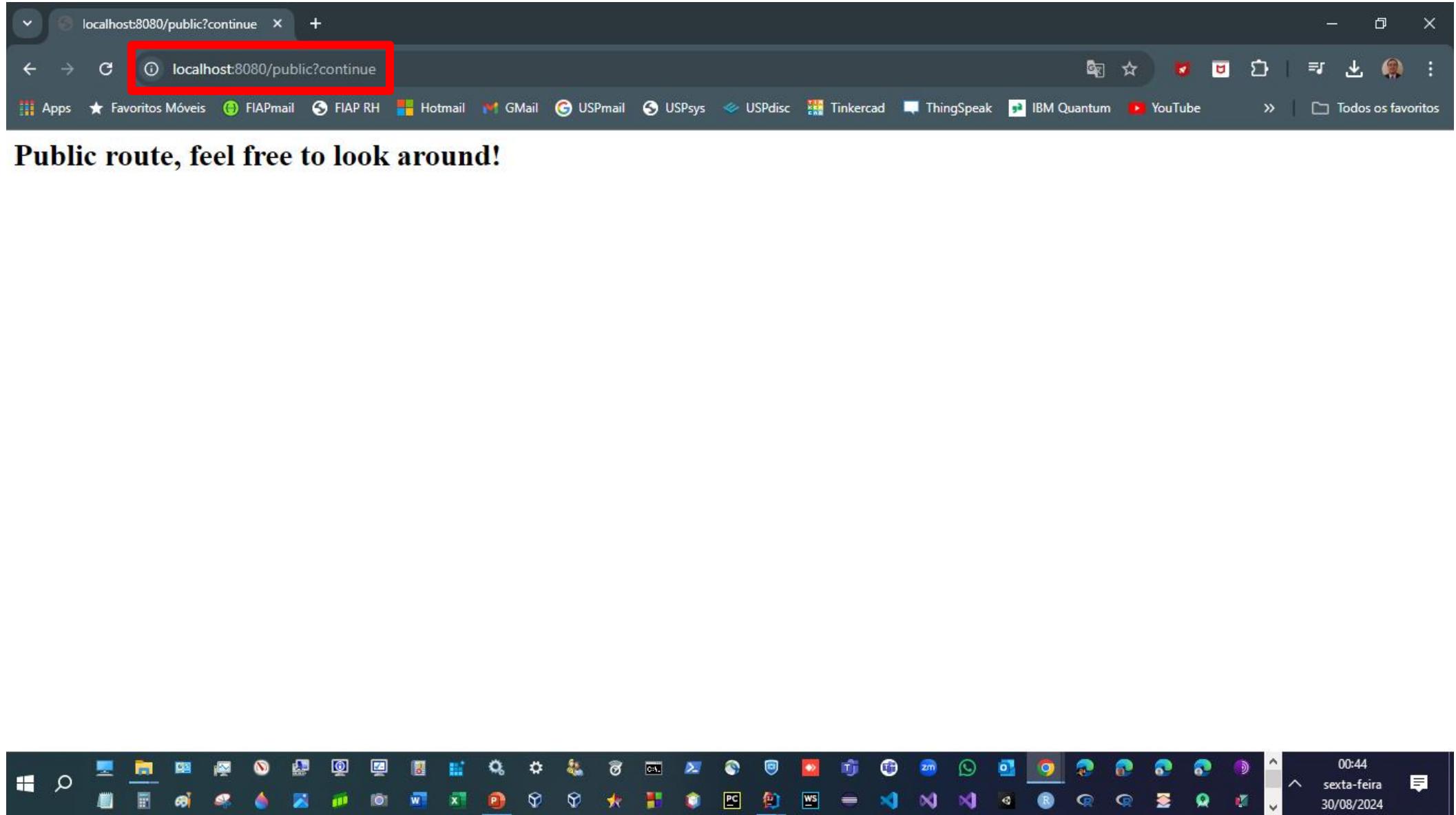
The screenshot shows a Java IDE interface with the following details:

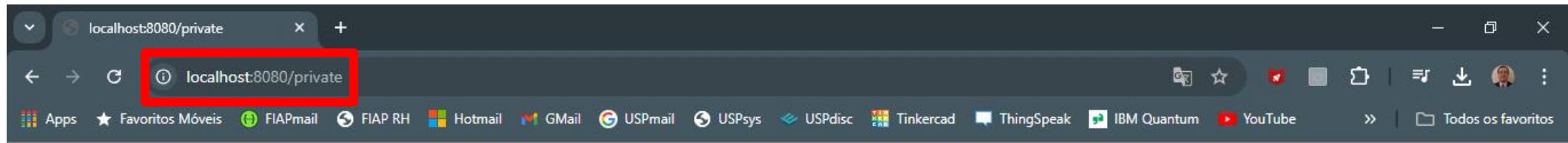
- Project Explorer:** Shows the project structure under "SecurityBasic".
- Code Editor:** Displays `SecurityBasicApplication.java` and `HttpController.java`.
- Terminal:** Shows the output of running the application. A red box highlights the line: "Using generated security password: 960a36f0-3243-4767-aad7-3689bb8a734c". Below it, a message states: "This generated password is for development use only. Your security configuration must be updated before running your application in production."

Copie o password fornecido para o login

```
2024-08-30T00:39:45.446-03:00 [WARNING] 3060 --- [SecurityBasic] [ restartedMain] .s.s.UserDetailsServiceAutoConfiguration :  
Using generated security password: 960a36f0-3243-4767-aad7-3689bb8a734c  
This generated password is for development use only. Your security configuration must be updated before running your application in production.
```



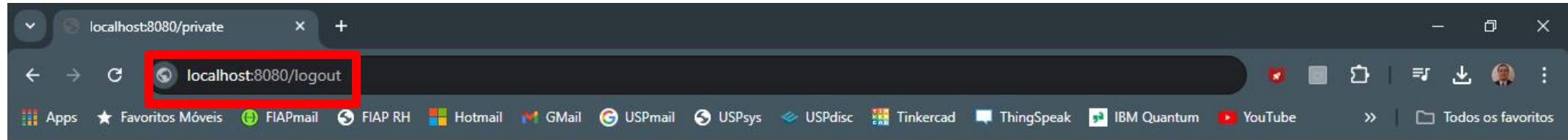




Private route, only authorized by Tranquilo!

O *Spring Security* já mantém logado.

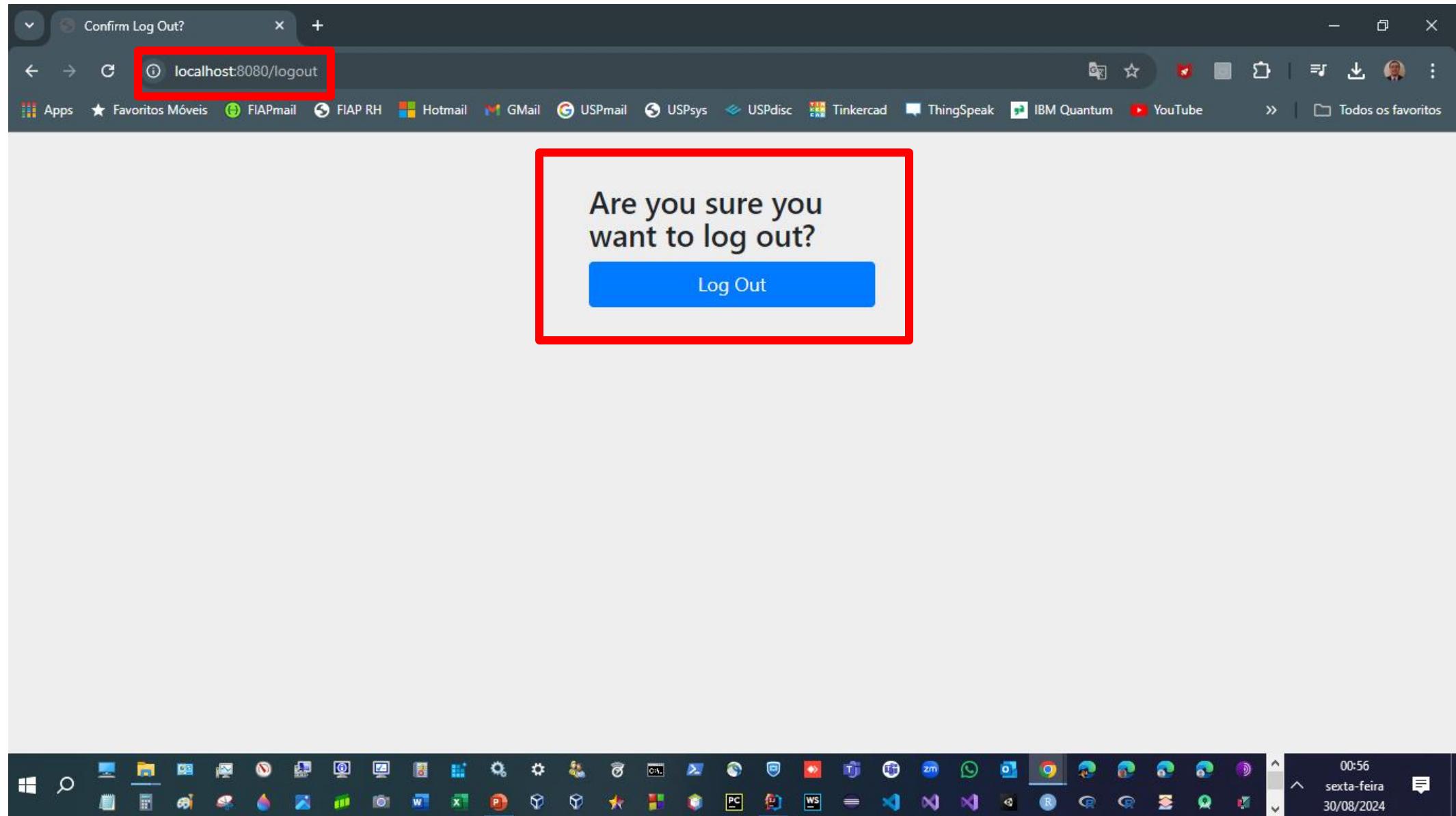


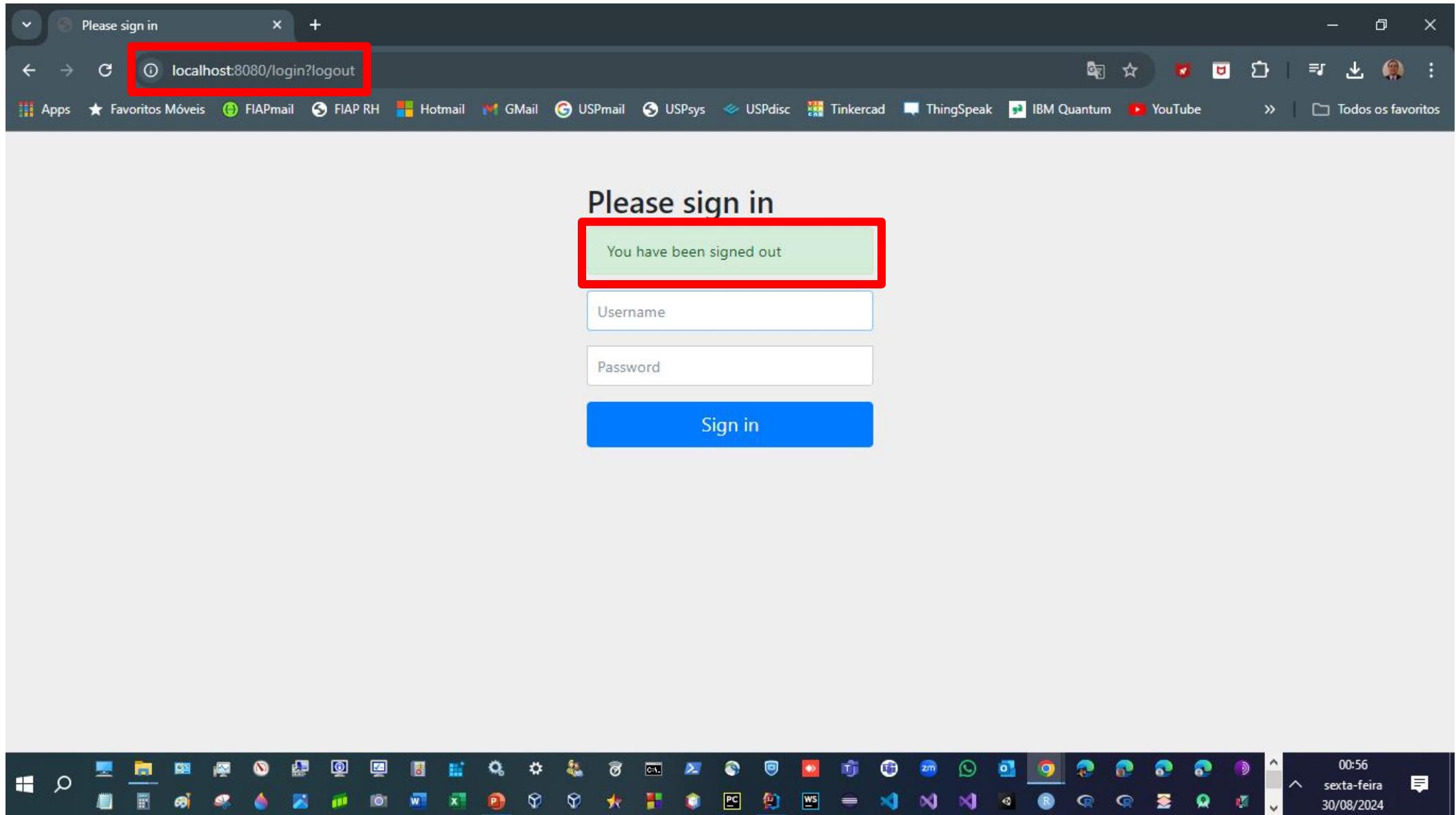


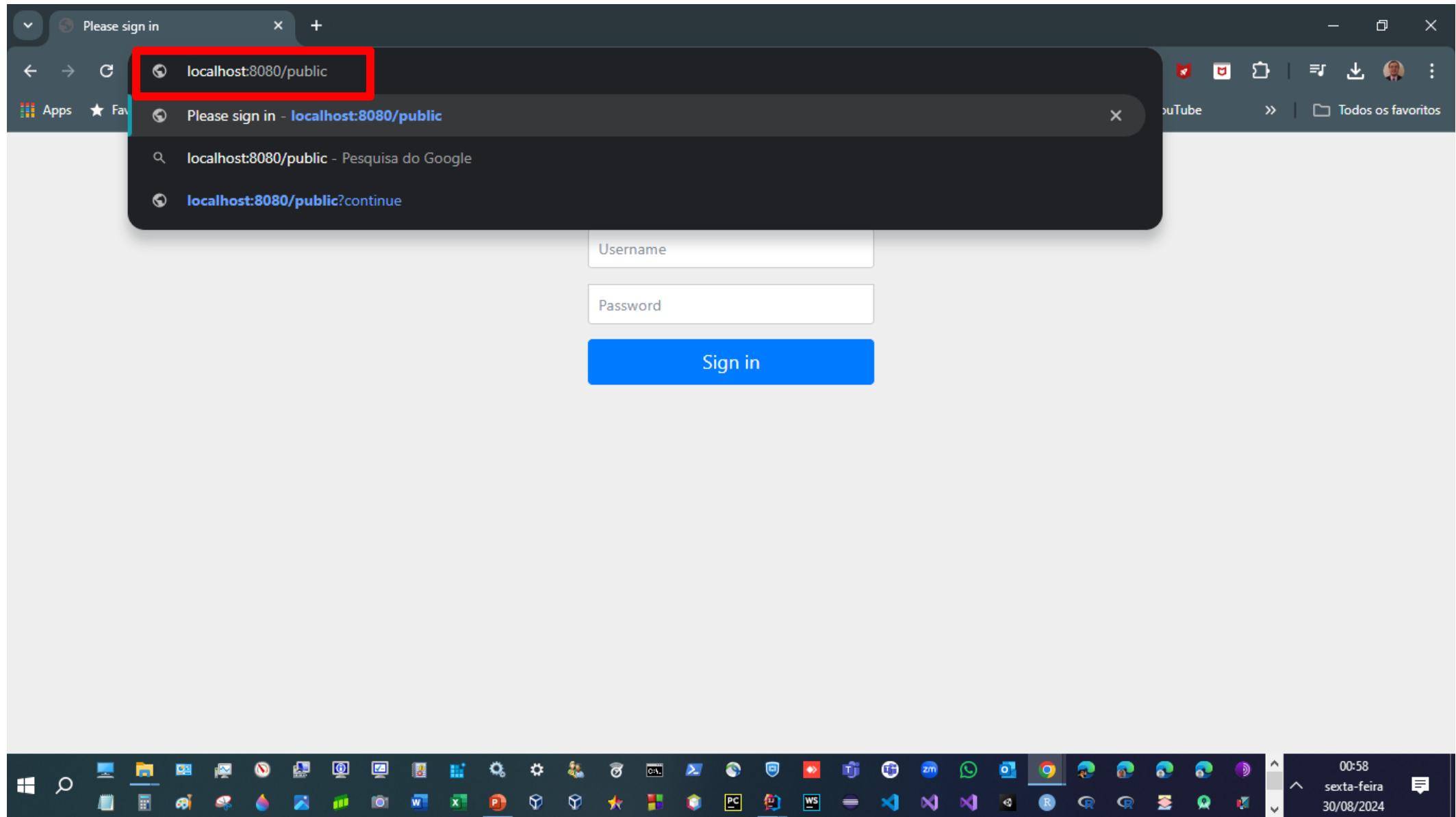
Private route, only authorized by Tranquilo!

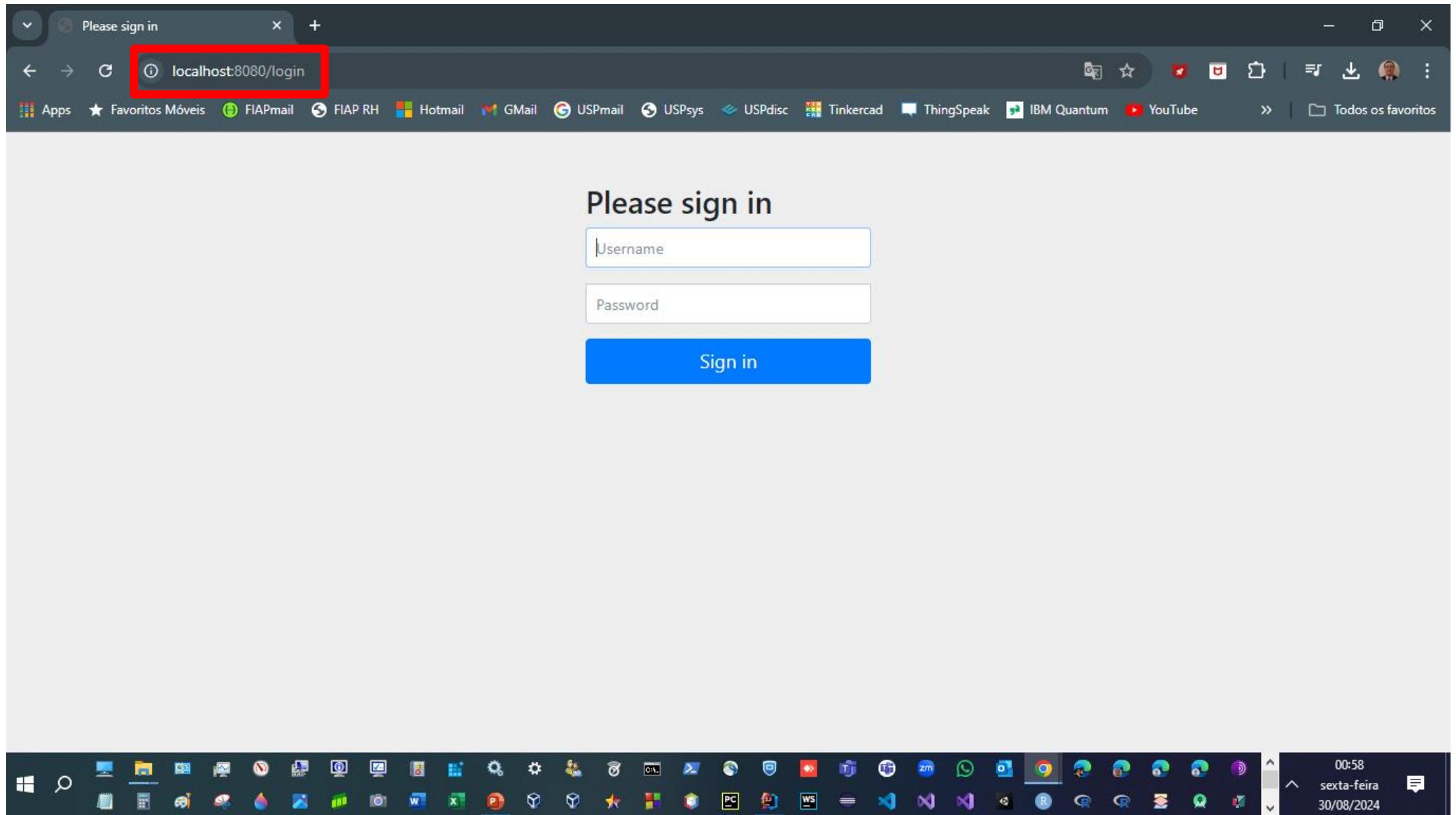
Dá para fazer *logout* com o *Spring Security*.











Como deixar a rota pública sem proteção padrão do *Spring Security*

Spring Security

- Rotas com necessidade de acesso diferentes

Caso em que queremos deixar a rota pública com acesso para qualquer pessoa, mas temos a intenção de usar o *Spring Security* para restringir a rota privada.

Desta forma, temos que realizar uma configuração diferente.

Devemos criar uma classe de configuração do *Spring* (*SecurityConfig.java*).

The screenshot shows an IDE interface with the following details:

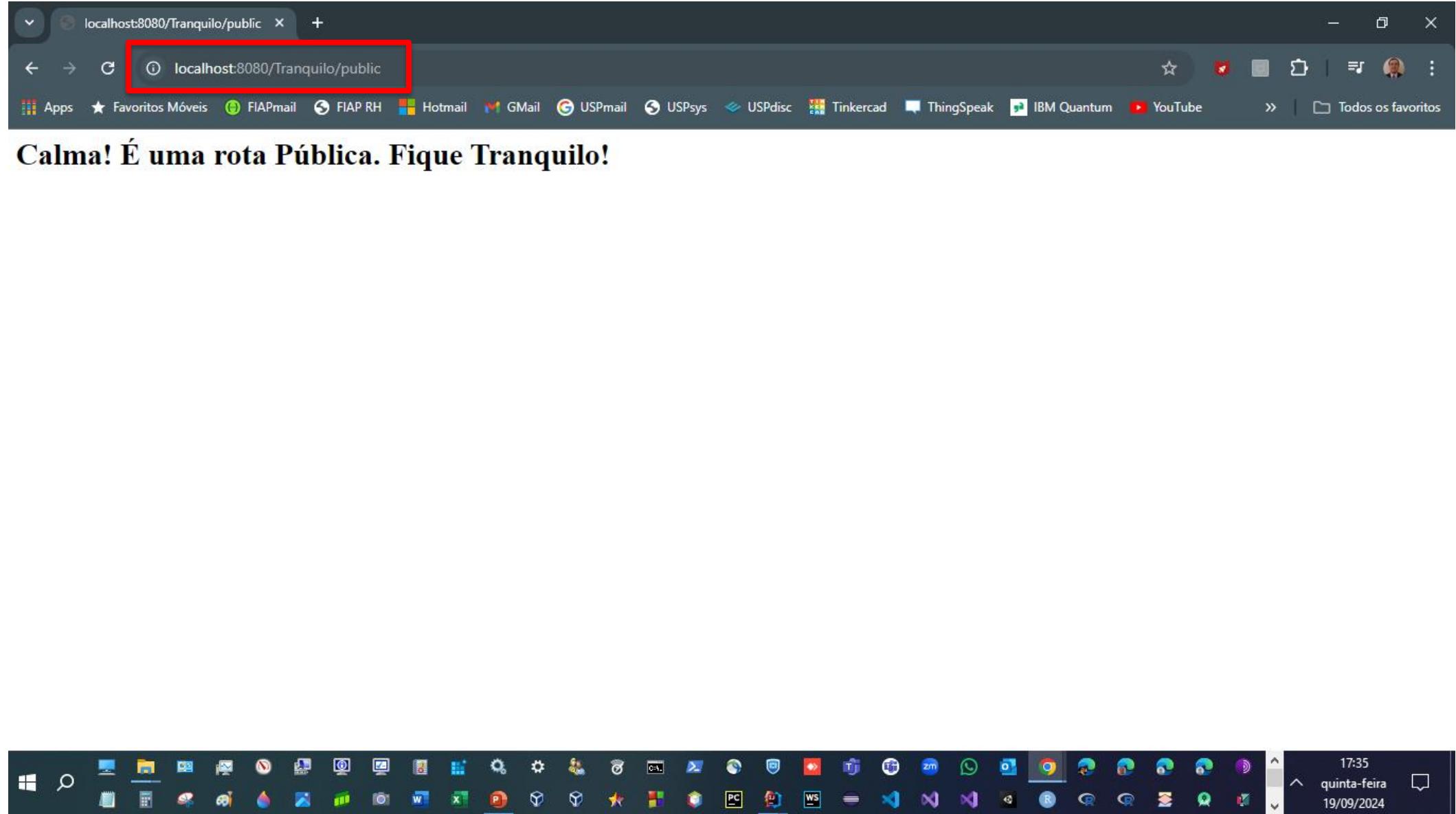
- Project View:** Shows the project structure under "SecurityImplement [SecurityBasic]". A red box highlights the package `br.com.fiap.SecurityBasic` and the class `SecurityConfig`.
- Code Editor:** Displays the `SecurityConfig.java` file. The code includes annotations `@Configuration` and `@EnableWebSecurity`, which are highlighted in yellow. The code also defines a `SecurityFilterChain` and a `formLogin` method.
- Annotations:** A callout box with the text "Anotação que habilita o *Spring Web Security*" points to the `@EnableWebSecurity` annotation.
- Status Bar:** Shows the file path: `SecurityImplement > src > main > java > br > com > fiap > SecurityBasic > SecurityConfig > securityFilterChain`, and the status: `26:26 CRLF UTF-8 4 spaces`.
- Taskbar:** Shows various system icons and the date/time: `17:39 quinta-feira 19/09/2024`.

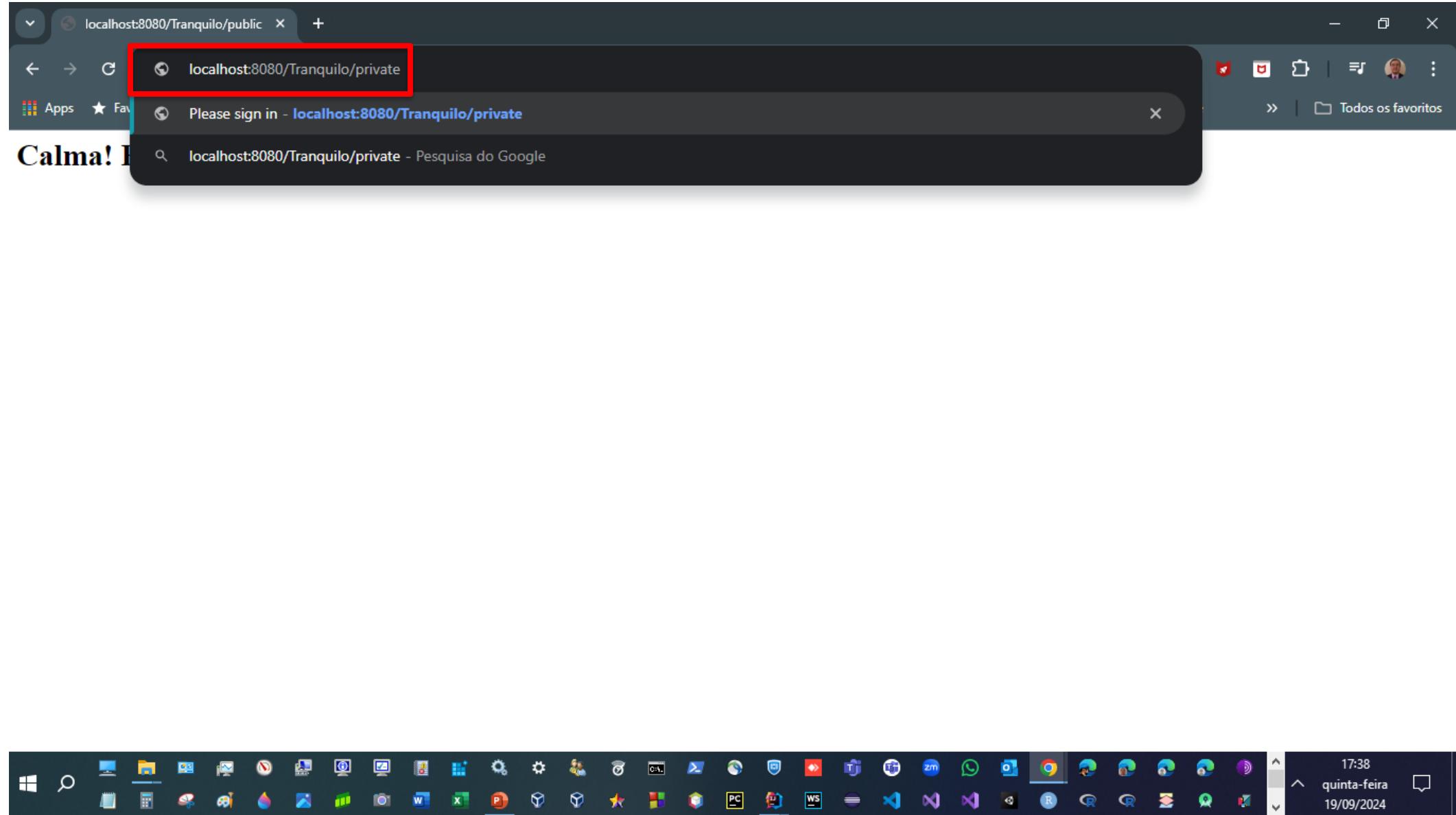
```
package br.com.fiap.SecurityBasic;

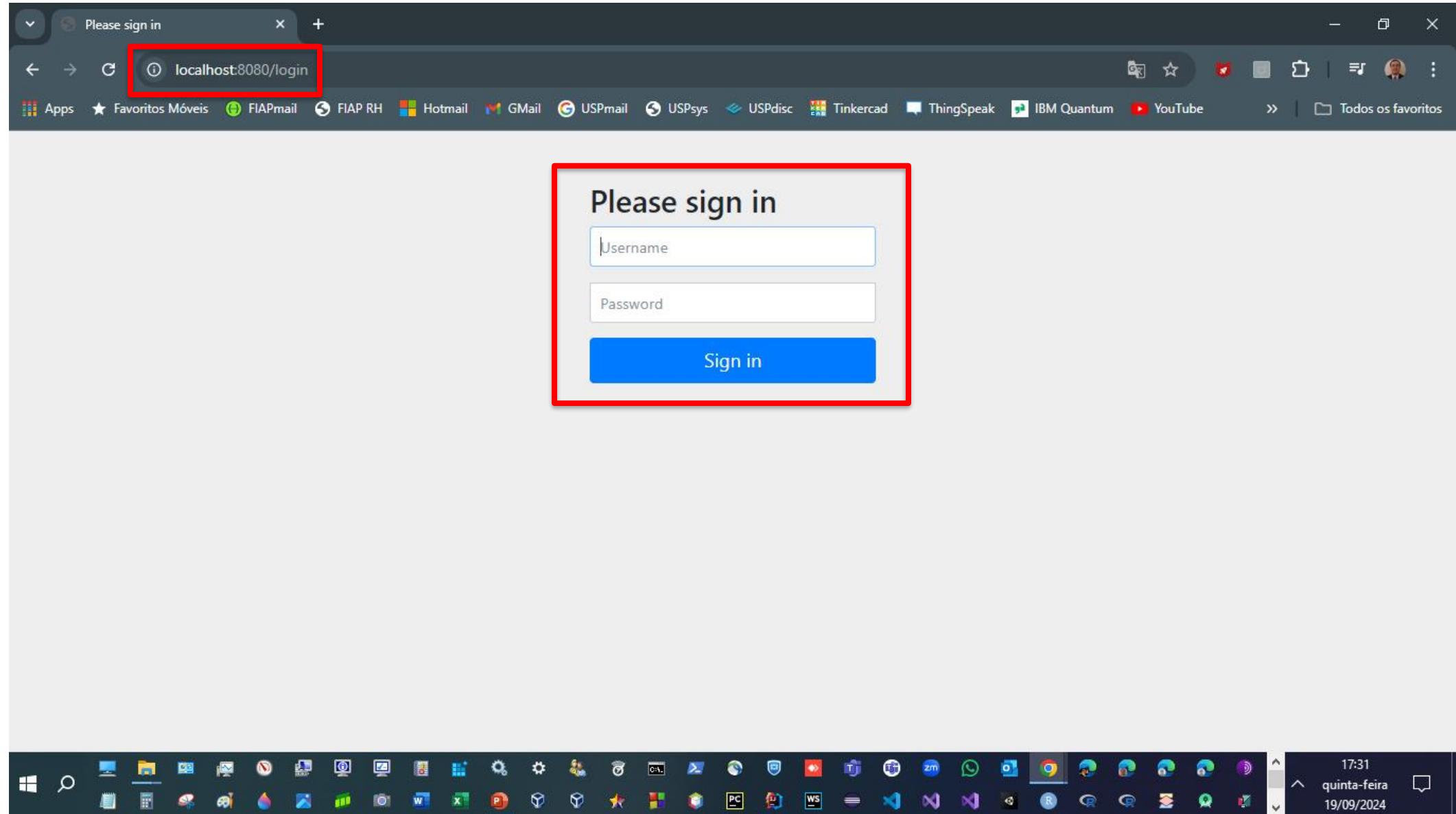
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.security.config.Customizer;
import org.springframework.security.config.annotation.web.builders.HttpSecurity;
import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
import org.springframework.security.web.SecurityFilterChain;

@Configuration
@EnableWebSecurity
public class SecurityConfig {

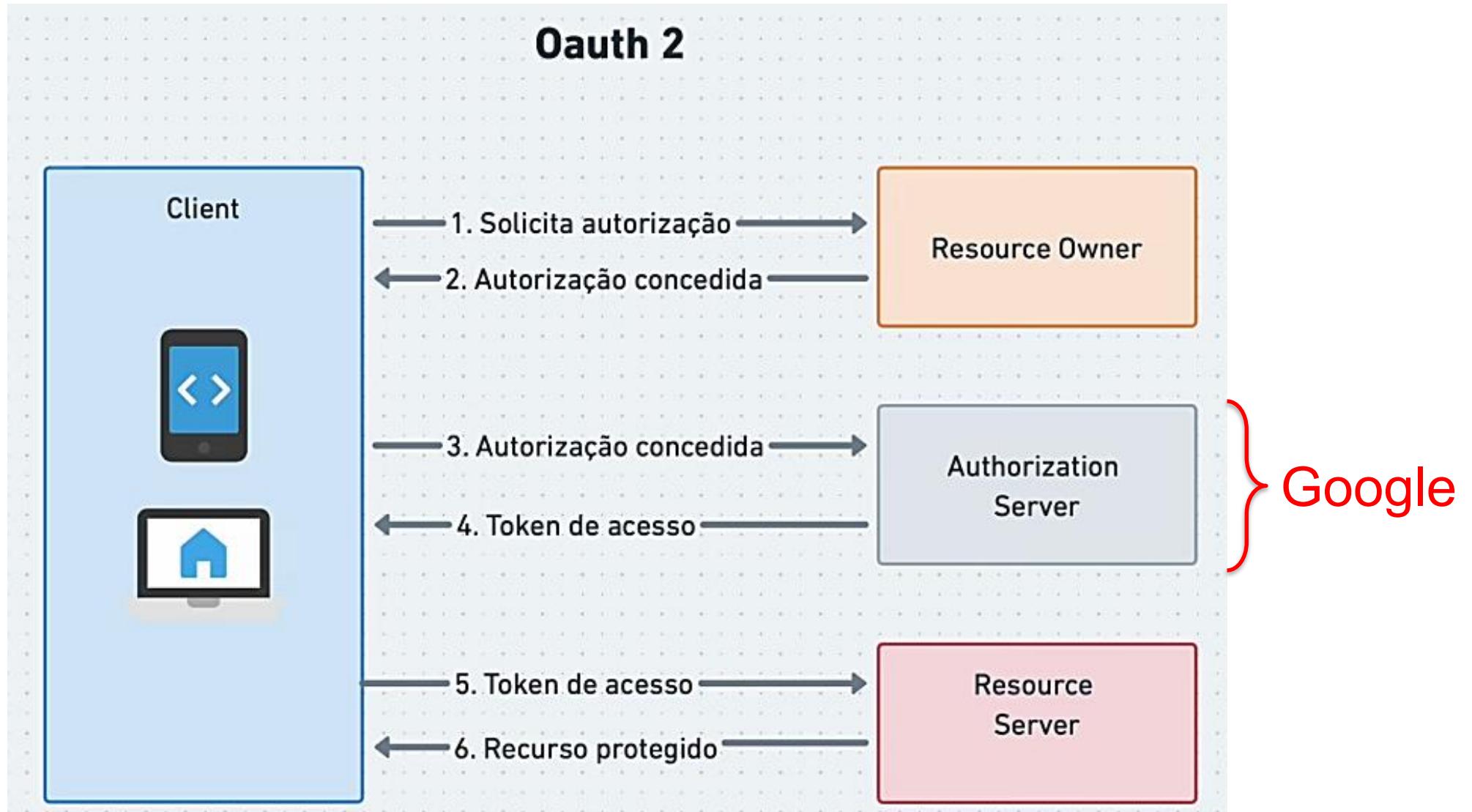
    // Filtros (Filters) - intercepta a requisicao e checa se e autenticada ou nao.
    // Cadeia de filtros de segurança. HttpSecurity e o objeto para configurar a segurança.
    @Bean
    SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {
        return http
            .authorizeHttpRequests(
                authorizeConfig -> {
                    authorizeConfig.requestMatchers("/Tranquilo/public").permitAll();
                    authorizeConfig.requestMatchers("/logout").permitAll();
                    authorizeConfig.anyRequest().authenticated();
                }
            )
            .formLogin(Customizer.withDefaults());
    }
}
```

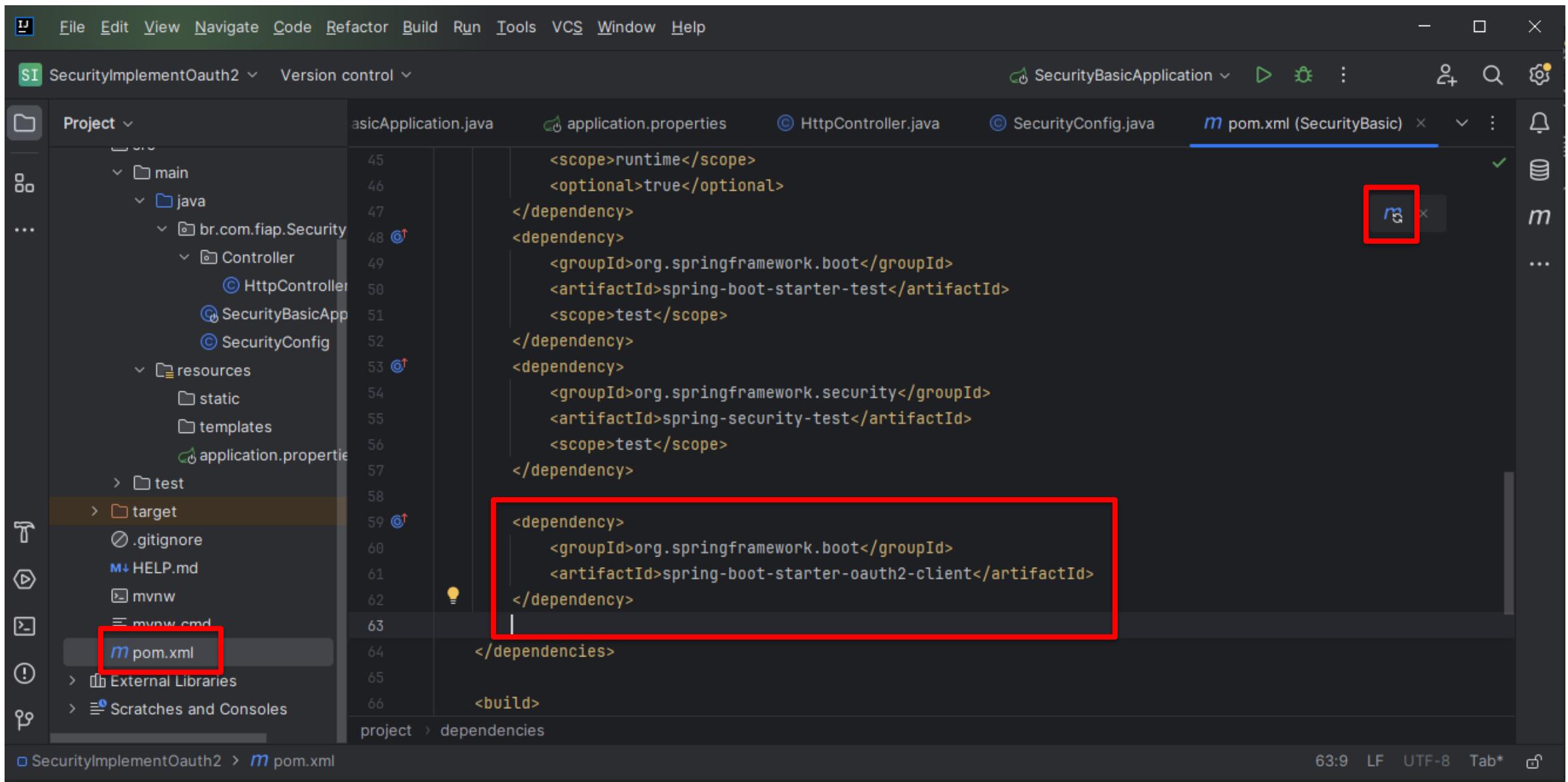






Oauth2 (Usando uma credencial provida pelo Google)





The screenshot shows an IDE interface with the following details:

- File Menu:** File, Edit, View, Navigate, Code, Refactor, Build, Run, Tools, VCS, Window, Help.
- Project Explorer:** Shows the project structure under "SecurityImplementOauth2". The "pom.xml" file is highlighted with a red box in the left sidebar.
- Code Editor:** The "pom.xml" file is open, displaying the following code snippet:

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-oauth2-client</artifactId>
</dependency>
```

A red box highlights the entire dependency block. The "pom.xml (SecurityBasic)" tab is also highlighted with a red box at the top right of the editor.

The screenshot shows a browser window with the URL start.spring.io. The page is a configuration tool for creating Spring Boot projects. It includes fields for Project (Gradle - Groovy, Gradle - Kotlin, Maven), Language (Java, Kotlin, Groovy), and Spring Boot version (3.4.0 (SNAPSHOT), 3.4.0 (M3), 3.3.5 (SNAPSHOT), 3.3.4, 3.2.11 (SNAPSHOT), 3.2.10). The 'Project Metadata' section contains fields for Group (com.example), Artifact (demo), Name (demo), Description (Demo project for Spring Boot), Package name (com.example.demo), and Packaging (Jar, War). Below these are Java version options (23, 21, 17). On the right, the 'Dependencies' section lists Spring Web (WEB), Spring Boot DevTools (DEVELOPER TOOLS), Spring Security (SECURITY), and OAuth2 Client (SECURITY). The 'OAuth2 Client' section is highlighted with a yellow box. At the bottom are buttons for GENERATE (CTRL + ⌘), EXPLORE (CTRL + SPACE), and SHARE... . The browser's address bar shows 'start.spring.io'. The taskbar at the bottom includes icons for various Windows applications like File Explorer, Edge, and Mail.

Spring Security OAuth2

- Vamos renomear o *application.properties* para *application.yaml*

YAML é uma linguagem de serialização de dados legível por humanos que é frequentemente usada para escrever arquivos de configuração. Dependendo de quem você perguntar, YAML significa *yet another markup language* ou *YAML ain't markup language* (uma sigla recursiva), o que enfatiza que YAML é para dados, não documentos.

YAML é uma linguagem de programação popular porque foi projetada para ser fácil de ler e entender. Também pode ser usada em conjunto com outras linguagens de programação.

Os arquivos YAML usam uma extensão *.yml* ou *.yaml* e seguem regras de sintaxe específicas.

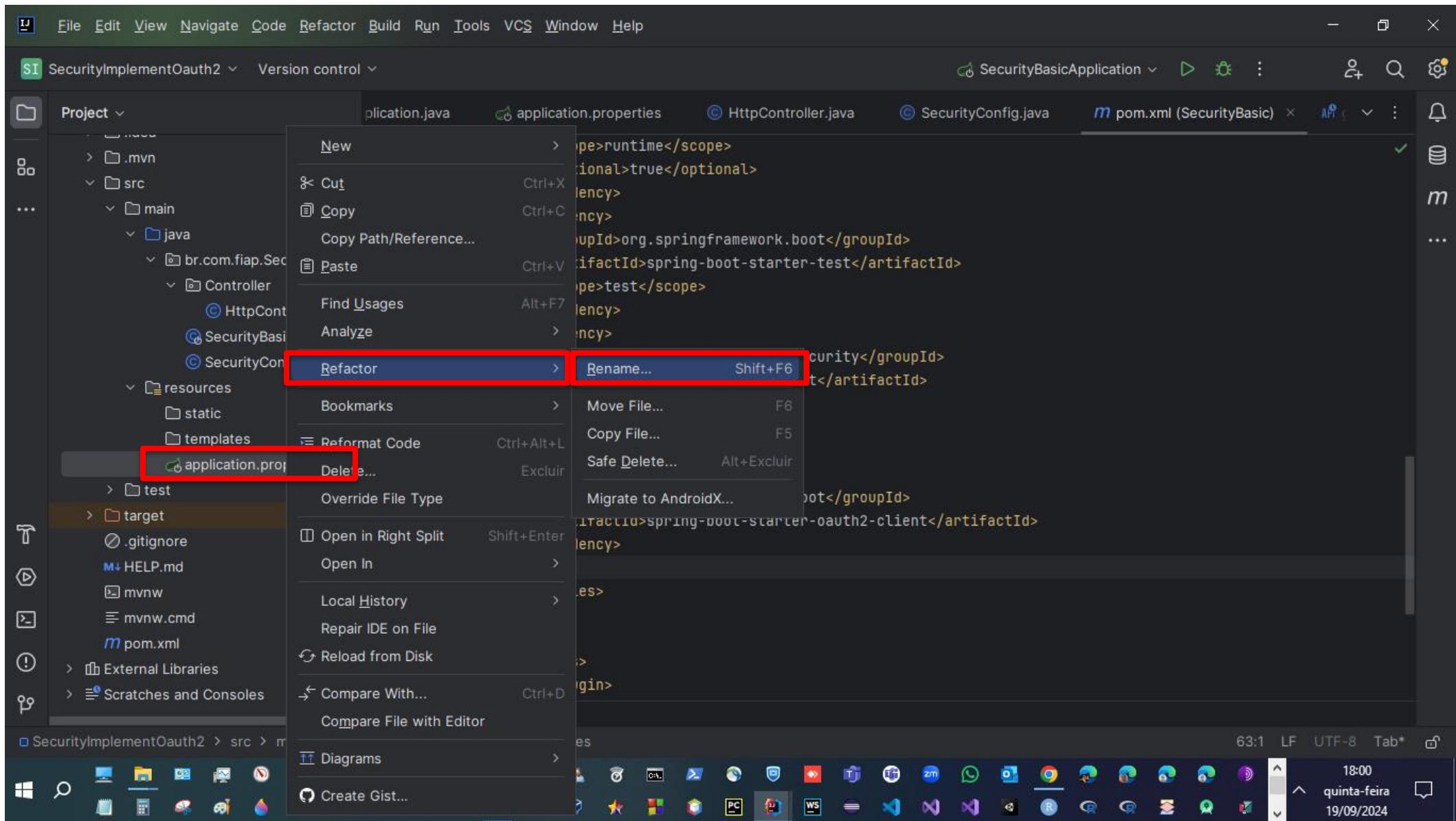
YAML tem recursos que vêm de Perl, C, XML, HTML e outras linguagens de programação. YAML também é um superconjunto de JSON, então arquivos JSON são válidos em YAML.

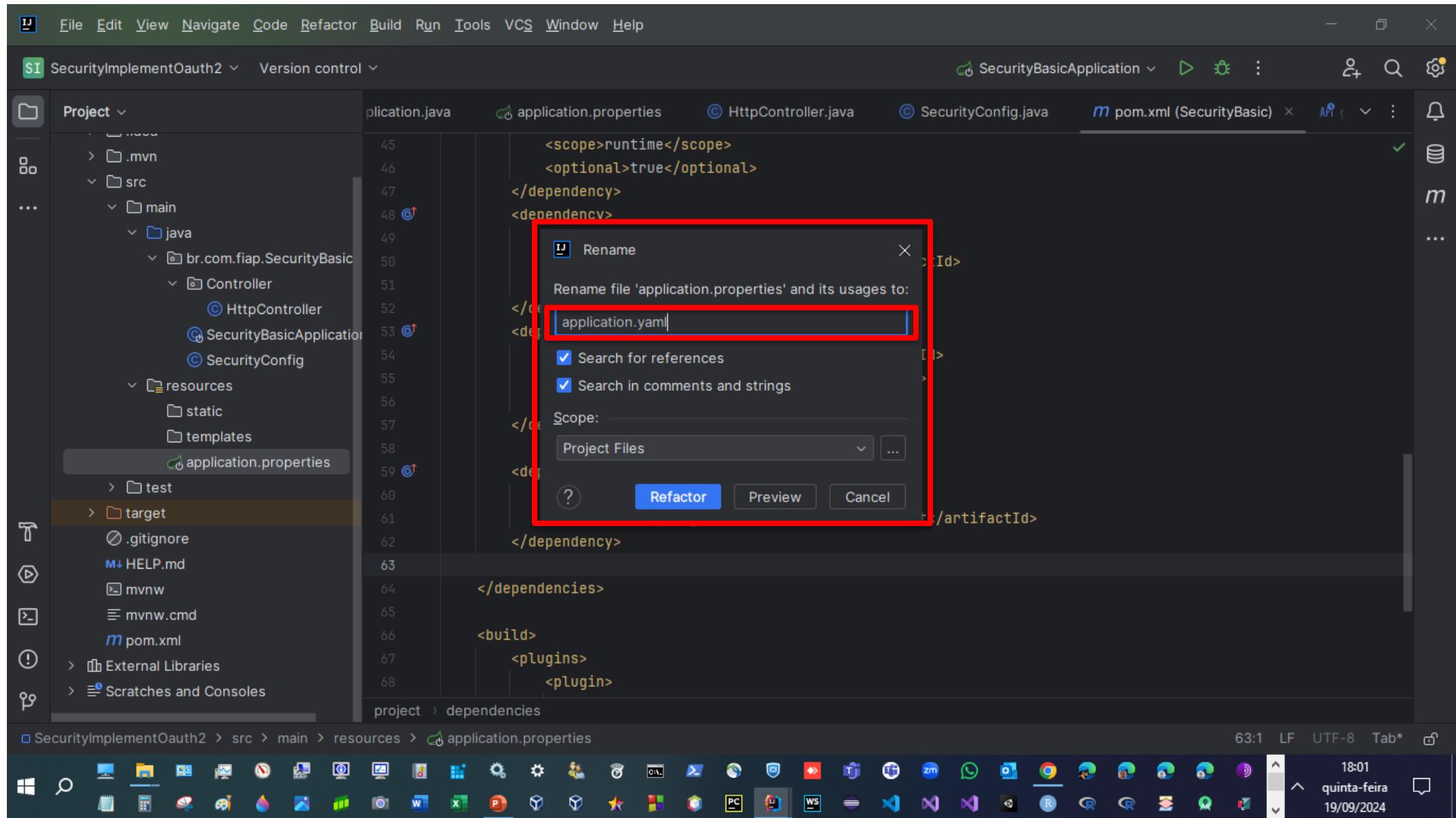
The screenshot shows a Java development environment with the following details:

- File Menu:** File, Edit, View, Navigate, Code, Refactor, Build, Run, Tools, VCS, Window, Help.
- Project Bar:** SecurityImplementOauth2, Version control.
- Toolbars:** Standard toolbar with icons for file operations, search, and navigation.
- Left Sidebar:** Project tree showing the project structure:
 - .mvn
 - src
 - main
 - java
 - br.com.fiap.SecurityBasic
 - Controller
 - HttpController
 - SecurityBasicApplication
 - SecurityConfig
 - resources
 - static
 - templates
 - test
 - target
 - .gitignore
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml
 - External Libraries
 - Scratches and Consoles
- Central Area:** Code editor showing the content of application.properties. The file is highlighted with a red box.

```
application.properties
```

```
45 <dependency>
46     <groupId>org.springframework.boot</groupId>
47     <artifactId>spring-boot-starter-test</artifactId>
48     <scope>test</scope>
49 </dependency>
50 <dependency>
51     <groupId>org.springframework.security</groupId>
52     <artifactId>spring-security-test</artifactId>
53     <scope>test</scope>
54 </dependency>
55 <dependency>
56     <groupId>org.springframework.boot</groupId>
57     <artifactId>spring-boot-starter-oauth2-client</artifactId>
58 </dependency>
59 </dependencies>
60 <build>
61     <plugins>
62         <plugin>
```
- Bottom Status Bar:** Shows the current file path (SecurityImplementOauth2 > src > main > resources > application.properties), code editor settings (63:1, LF, UTF-8, Tab*), and system status (18:00, quint-feira, 19/09/2024).





The screenshot shows a Java IDE interface with the following details:

- Project Explorer:** Shows the project structure with files like `application.java`, `HttpController.java`, `SecurityConfig.java`, and `pom.xml`. The `application.yaml` file is highlighted with a red box.
- Code Editor:** Displays the content of the `application.yaml` file, which includes Spring Boot dependencies for OAuth2 Client and Security.
- Toolbar:** Includes standard icons for file operations, navigation, and tools.
- Status Bar:** Shows the current file path (`SecurityImplementOauth2 > src > main > resources > application.properties`), encoding (`UTF-8`), and system information (`63:1 LF`).
- System Tray:** Shows the date and time (`18:01 quinta-feira 19/09/2024`).

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-oauth2-client</artifactId>
    <scope>runtime</scope>
    <optional>true</optional>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>org.springframework.security</groupId>
    <artifactId>spring-security-test</artifactId>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-oauth2-client</artifactId>
    </dependency>
</dependencies>
<build>
    <plugins>
        <plugin>
```

The screenshot shows a Java IDE interface with the following details:

- Project:** SecurityImplementOauth2
- File:** application.yaml
- Content:**

```
spring:
  security:
    oauth2:
      client:
        registration:
          google:
            client-id: <adicone aqui>
            client-secret: <adicone aqui>
```

- IDE UI Elements:**
 - Toolbar: File, Edit, View, Navigate, Code, Refactor, Build, Run, Tools, VCS, Window, Help.
 - Search bar: Version control, SecurityBasicApplication.
 - Project Explorer: Shows the project structure with .mvn, src (main.java, resources), target, .gitignore, HELP.md, mvnw, mvnw.cmd, pom.xml, External Libraries, and Scratches and Consoles.
 - Code Editor: Shows the application.yaml file with syntax highlighting for YAML keywords and comments.
 - Bottom Status Bar: SecurityImplementOauth2 > src > main > resources > application.yaml, 9:1 LF ISO-8859-1 2 spaces, 18:49 quinta-feira 19/09/2024.
 - Bottom Taskbar: Shows various application icons.

Spring Security OAuth2

- Devemos agora criar as credenciais Google

Para nos tornarmos “parceiros” do Google e criar as credenciais para acesso, facilitando a troca de *tokens*, basta criar as credenciais em:

<https://console.cloud.google.com/apis/credentials>

Daí é só seguir os passos de nova credencial.

The screenshot shows the Google Cloud Platform API Credentials page. At the top, there's a banner for a free trial with US\$ 300 credit. The main navigation bar includes links for Apps, Favoritos Móveis, FIAPmail, FIAP RH, Hotmail, Gmail, USPmail, USPsys, USPdisc, Tinkercad, ThingSpeak, IBM Quantum, YouTube, and Todos os favoritos. The user profile icon is highlighted with a red box.

The left sidebar has sections for APIs e serviços, APIs e serviços ativados, Biblioteca, Credenciais (which is selected and highlighted with a blue box), Tela de permissão OAuth, and Contratos de uso de página.

The main content area is titled "Credenciais" and features a "Criar Credenciais" button, which is also highlighted with a red box. Below it, there's a note about creating credentials for active APIs and a link to "CONFIGURAR TELA DE CONSENTIMENTO".

Under "Chaves de API", it says "Nenhuma chave de API a ser exibida".

Under "IDs do cliente OAuth 2.0", it says "Nenhum cliente do OAuth a ser exibido".

Under "Contas de serviço", there's a link to "Gerenciar contas de serviço".

The bottom taskbar shows various Windows icons and the system clock indicating 18:13 on a Thursday, 19/09/2024.

RPI Credenciais – APIs e serviços – +

console.cloud.google.com/apis/credentials?pli=1&project=nodal-keyword-287818

Apps Favoritos Móveis FIAPmail FIAP RH Hotmail GMail USPmail USPsys USPdisc Tinkercad ThingSpeak IBM Quantum YouTube Todos os favoritos

Comece seu teste gratuito com US\$ 300 em crédito. Não se preocupe, você não será cobrado se seus créditos acabarem. [Saiba mais](#)

DISPENSAR COMECE GRATUITAMENTE

Google Cloud My First Project Pesquise (/) recursos, documentos, produtos e muito mais Pesquisa

API APIs e serviços Credenciais + CRIAR CREDENCIAIS EXCLUIR RESTAURAR CREDENCIAIS EXCLUÍDAS

APIs e serviços ativados Biblioteca Credenciais Tela de permissão OAuth Contratos de uso de página

Chave de API Crie credenciais para seu projeto Identifica seu projeto usando uma chave de API simples para verificar cota e limite.

Lembre-se de que é importante gerenciar suas credenciais de forma segura.

ID do cliente OAuth Solicita a permissão do usuário para que o aplicativo possa acessar os dados desse usuário.

Chaves de API

Nome Nenhuma chave de API encontrada.

CONFIGURAR TELA DE CONSENTIMENTO

Contas de serviço

E-mail Nome ↑ Ações

Ajude-me a escolher Faz algumas perguntas para ajudar você a decidir que tipo de credencial usar.

IDs do cliente OAuth 2.0

Nome Data da criação ↓ Tipo ID do cliente Ações

Nenhum cliente do OAuth a ser exibido.

Gerenciar contas de serviço

https://console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fpli%3D1%26project%3Dnodal-keyword-287818&project=nodal-keyword-287818

18:17 quinta-feira 19/09/2024

The screenshot shows the Google Cloud Platform API Credentials page. The left sidebar has 'Credenciais' selected. The main area shows three options under 'Credenciais': 'Chave de API', 'ID do cliente OAuth', and 'Ajude-me a escolher'. The 'ID do cliente OAuth' option is highlighted with a red box. Below it is a section for 'Chaves de API' which is currently empty. At the bottom, there's a table for 'IDs do cliente OAuth 2.0' which also shows no results. The status bar at the bottom indicates the URL: https://console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fpli%3D1%26project%3Dnodal-keyword-287818&project=nodal-keyword-287818 and the date and time: 18:17 quinta-feira 19/09/2024.

The screenshot shows a browser window with the URL <https://console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fpli%3D1%26project%3Dnodal-keyword-287818...>. The page is titled "API Criar ID do cliente do OAuth". The left sidebar has "APIs e serviços" selected, with "Credenciais" highlighted. A message box states: "Para criar um ID do cliente OAuth, primeiro é preciso configurar sua tela de permissão" and contains a "CONFIGURAR TELA DE CONSENTIMENTO" button, which is highlighted with a red box. The top navigation bar includes links for "Google Cloud", "My First Project", "Pesquisa", "Pesquisa", and user profile.

RPI APIs e serviços – My First Project

console.cloud.google.com/apis/credentials/consent?project=nodal-keyword-287818

Apps Favoritos Móveis FIAPmail FIAP RH Hotmail GMail USPmail USPsys USPdisc Tinkercad ThingSpeak IBM Quantum YouTube Todos os favoritos

Comece seu teste gratuito com US\$ 300 em crédito. Não se preocupe, você não será cobrado se seus créditos acabarem. [Saiba mais](#)

DISPENSAR COECE GRATUITAMENTE

Google Cloud My First Project Pesquise (/) recursos, documentos, produtos e muito mais Pesquisa

API APIs e serviços Tela de permissão OAuth

APIs e serviços ativados Escolha como você quer configurar e registrar seu aplicativo, incluindo os usuários-alvo. É possível associar somente um aplicativo ao seu projeto.

Biblioteca

Credenciais

Tela de permissão OAuth

Contratos de uso de página

User Type

Interno [Saiba mais sobre o tipo de usuário](#)

Externo [Saiba mais sobre o tipo de usuário](#)

Disponível para qualquer usuário de teste que tenha uma Conta do Google. O app ficará no modo de testes e só será disponibilizado aos usuários que você adicionou à lista de usuários de teste. Quando o app estiver pronto para ser enviado à produção, talvez seja necessário verificá-lo. [Saiba mais sobre o tipo de usuário](#)

CRIAR

Dê sua opinião sobre nossa experiência com o OAuth

Saiba mais

Tela de consentimento do Google OAuth

O que é a tela de consentimento do OAuth?

O que são os escopos de consentimento do OAuth?

O que são escopos confidenciais da API?

O que são escopos restritos da API?

O processo de registro de app

Preciso de que informações?

Meu app precisa ser verificado pelo Google?

O que acontecerá se eu não verificar meu app?

18:21 quinta-feira 19/09/2024

The screenshot shows the Google Cloud Platform's API Permissions page for a project named 'My First Project'. On the left, there's a sidebar with navigation links like 'APIs e serviços', 'APIs e serviços ativados', 'Biblioteca', 'Credenciais', and 'Tela de permissão OAuth'. The 'Tela de permissão OAuth' link is highlighted with a red box. The main content area is titled 'Tela de permissão OAuth' and contains a section for 'User Type'. It offers two options: 'Interno' (selected by default) and 'Externo' (selected and highlighted with a red box). Below each option is a description and a link to 'Saiba mais sobre o tipo de usuário'. At the bottom of this section is a large blue 'CRIAR' button. To the right, there's a sidebar with several questions: 'Tela de consentimento do Google OAuth', 'O que é a tela de consentimento do OAuth?', 'O que são os escopos de consentimento do OAuth?', 'O que são escopos confidenciais da API?', 'O que são escopos restritos da API?', 'O processo de registro de app', 'Preciso de que informações?', 'Meu app precisa ser verificado pelo Google?', and 'O que acontecerá se eu não verificar meu app?'. The top of the page shows the browser's address bar with 'console.cloud.google.com/apis/credentials/consent?project=nodal-keyword-287818' and various browser icons. The bottom of the page shows the Windows taskbar with icons for various applications like File Explorer, Microsoft Edge, and Microsoft Word.

RPI APIs e serviços – My First Project

console.cloud.google.com/apis/credentials/consent?project=nodal-keyword-287818

Apps Favoritos Móveis FIAPmail FIAP RH Hotmail GMail USPmail USPsys USPdisc Tinkercad ThingSpeak IBM Quantum YouTube Todos os favoritos

Comece seu teste gratuito com US\$ 300 em crédito. Não se preocupe, você não será cobrado se seus créditos acabarem. [Saiba mais](#)

DISPENSAR **COMECE GRATUITAMENTE**

Google Cloud My First Project Pesquise (/) recursos, documentos, produtos e muito mais Pesquisa

API APIs e serviços

- APIs e serviços ativados
- Biblioteca
- Credenciais
- Tela de permissão OAuth**
- Contratos de uso de página

Aplicativo da Web EDITAR APP

Status de publicação ?

Teste PUBLICAR APPLICATIVO

Tipo de usuário

Externo ? TORNAR INTERNO

Limite de usuários do OAuth ?

Enquanto o status de publicação mostrar a opção "Testando", apenas os usuários de teste conseguirão acessar o aplicativo. O limite de usuários

Saiba mais

Tela de consentimento do Google OAuth

O que é a tela de consentimento do OAuth?

O que são os escopos de consentimento do OAuth?

O que são escopos confidenciais da API?

O que são escopos restritos da API?

O processo de registro de app

Preciso de que informações?

Meu app precisa ser verificado pelo Google?

O que acontecerá se eu não verificar meu app?

18:50 quinta-feira 19/09/2024

A red box highlights the 'Tela de permissão OAuth' (OAuth permission screen) section, which contains the following configuration:

- Aplicativo da Web**: EDITAR APP
- Status de publicação**: Teste (PUBLICAR APPLICATIVO)
- Tipo de usuário**: Externo
- Limite de usuários do OAuth**: Unlabeled (described in text below as 'Testando')

The sidebar on the right lists several frequently asked questions related to the OAuth consent screen:

- O que é a tela de consentimento do OAuth?
- O que são os escopos de consentimento do OAuth?
- O que são escopos confidenciais da API?
- O que são escopos restritos da API?
- O processo de registro de app
- Preciso de que informações?
- Meu app precisa ser verificado pelo Google?
- O que acontecerá se eu não verificar meu app?

The screenshot shows the Google Cloud Platform API Credentials page. The URL in the browser is `console.cloud.google.com/apis/credentials?pli=1&project=nodal-keyword-287818`. The left sidebar has a red box around the 'Credenciais' link under the 'APIs e serviços' section. The main area shows sections for 'Chaves de API', 'IDs do cliente OAuth 2.0', and 'Contas de serviço'. A red box highlights the '+ CRIAR CREDENCIAIS' button at the top of the 'Credenciais' section. The status bar at the bottom shows system icons and the date/time: 18:13 quinta-feira 19/09/2024.

The screenshot shows a Microsoft Edge browser window with the URL console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fproject%3Dnodal-keyword-287818&project=no.... The page is titled "Criar ID do cliente do OAuth". The left sidebar shows "APIs e serviços" with "Credenciais" selected. A red box highlights the dropdown menu for "Tipo de aplicativo *", which contains the following options:

- Aplicativo da Web
- Android
- Extensão do Chrome
- iOS
- TVs e dispositivos de entrada limitados
- App para computador
- Plataforma Universal do Windows (UWP)

The browser's address bar shows the URL above, and the taskbar at the bottom includes icons for various applications like File Explorer, Microsoft Word, and Microsoft Excel.

The screenshot shows a browser window with the URL console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fproject%3Dnodal-keyword-287818&project=no.... The page is titled "API Criar ID do cliente do OAuth". The left sidebar shows "APIs e serviços" with "Credenciais" selected. The main content area is titled "Criar ID do cliente do OAuth". It explains that a client ID identifies a unique app on Google's OAuth servers. A dropdown menu for "Tipo de aplicativo" is set to "Aplicativo da Web". A red box highlights the "Nome" input field, which contains "teste-spring-security". Below the form, a note says "O nome do cliente é usado para nomear o cliente no console e não será mostrado aos usuários finais." A tooltip indicates that included domains will be added to the OAuth consent screen. At the bottom, there's a section for "Origens JavaScript autorizadas" with a "+ ADICIONAR URI" button. The taskbar at the bottom includes icons for various Windows applications like File Explorer, Edge, and PowerShell.

The screenshot shows a Microsoft Edge browser window with the URL console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fproject%3Dnodal-keyword-287818&project=no.... The page is titled "API Criar ID do cliente do OAuth". The left sidebar has "Credenciais" selected. The main content area shows "Origens JavaScript autorizadas" and "URIs de redirecionamento autorizados". A red box highlights the "+ ADICIONAR URI" button under "URIs de redirecionamento autorizados". The bottom right corner of the screen shows the Windows taskbar with the date and time (18:32 quinta-feira 19/09/2024).

API Criar ID do cliente do OAuth

console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fproject%3Dnodal-keyword-287818&project=no...

Apps Favoritos Móveis FIAPmail FIAP RH Hotmail GMail USPmail USPsys USPdisc Tinkercad ThingSpeak IBM Quantum YouTube Todos os favoritos

Comece seu teste gratuito com US\$ 300 em crédito. Não se preocupe, você não será cobrado se seus créditos acabarem. [Saiba mais](#)

DISPENSAR COMECE GRATUITAMENTE

Google Cloud My First Project Pesquise (/) recursos, documentos, produtos e muito mais Pesquisa

API APIs e serviços Criar ID do cliente do OAuth

APIs e serviços ativados Biblioteca Credenciais Tela de permissão OAuth Contratos de uso de página

Origens JavaScript autorizadas

Para usar com solicitações de um navegador

+ ADICIONAR URI

URIs de redirecionamento autorizados

Para usar com solicitações de um servidor da Web

+ ADICIONAR URI

Observação: pode levar de cinco minutos a algumas horas para que as configurações entrem em vigor

CRIAR CANCELAR

18:32 quinta-feira 19/09/2024

Spring Security Oauth2

- Devemos colocar agora a URI de redirecionamento autorizada

Para que exista o acesso com o redirecionamento, é preciso declarar a URI de redirecionamento na criação do ID Oauth:

<http://localhost:8080/login/oauth2/code/google>

Da URI, o trecho que é obrigatório para a conexão do Servidor de Autorização (Google), é:
/login/oauth2/code/google

The screenshot shows a web browser window for the Google Cloud Platform (GCP) API Credentials page. The URL in the address bar is `console.cloud.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fproject%3Dnodal-keyword-287818&project=no...`. The page title is "API Criar ID do cliente do OAuth".

The left sidebar menu is open, showing the following options under "API":

- APIs e serviços
- APIs e serviços ativados
- Biblioteca
- Credenciais** (selected)
- Tela de permissão OAuth
- Contratos de uso de página

The main content area displays the "Criar ID do cliente do OAuth" (Create OAuth client ID) form. It includes sections for "Origens Javascript autorizadas" (Authorized Javascript origins) and "URLs de redirecionamento autorizados".

In the "URLs de redirecionamento autorizados" section, a text input field contains the value `http://localhost:8080/login/oauth2/code/google`, which is highlighted with a red rectangle.

At the bottom of the form, there are two buttons: "CRIAR" (Create) and "CANCELAR" (Cancel). The "CRIAR" button is also highlighted with a red rectangle.

The browser's taskbar at the bottom shows various pinned icons, and the system tray indicates the date and time as "quinta-feira 19/09/2024" and "18:39".

The screenshot shows the Google Cloud Platform API Credentials page. A modal window titled "Cliente OAuth criado" (OAuth Client created) is displayed, highlighting the client details:

- ID do cliente: 724180968308-npmodk06cuer0ojnlbmq3kqalbuk7uj.apps.googleusercontent.com
- Chave secreta do cliente: (redacted)
- Data da criação: 19 de setembro de 2024 18:41:27 GMT-3
- Status: Ativada

At the bottom of the modal, there is a "OK" button. The entire modal is enclosed in a red box.

On the right side of the screen, a list of existing OAuth clients is visible, showing the same ID and status.

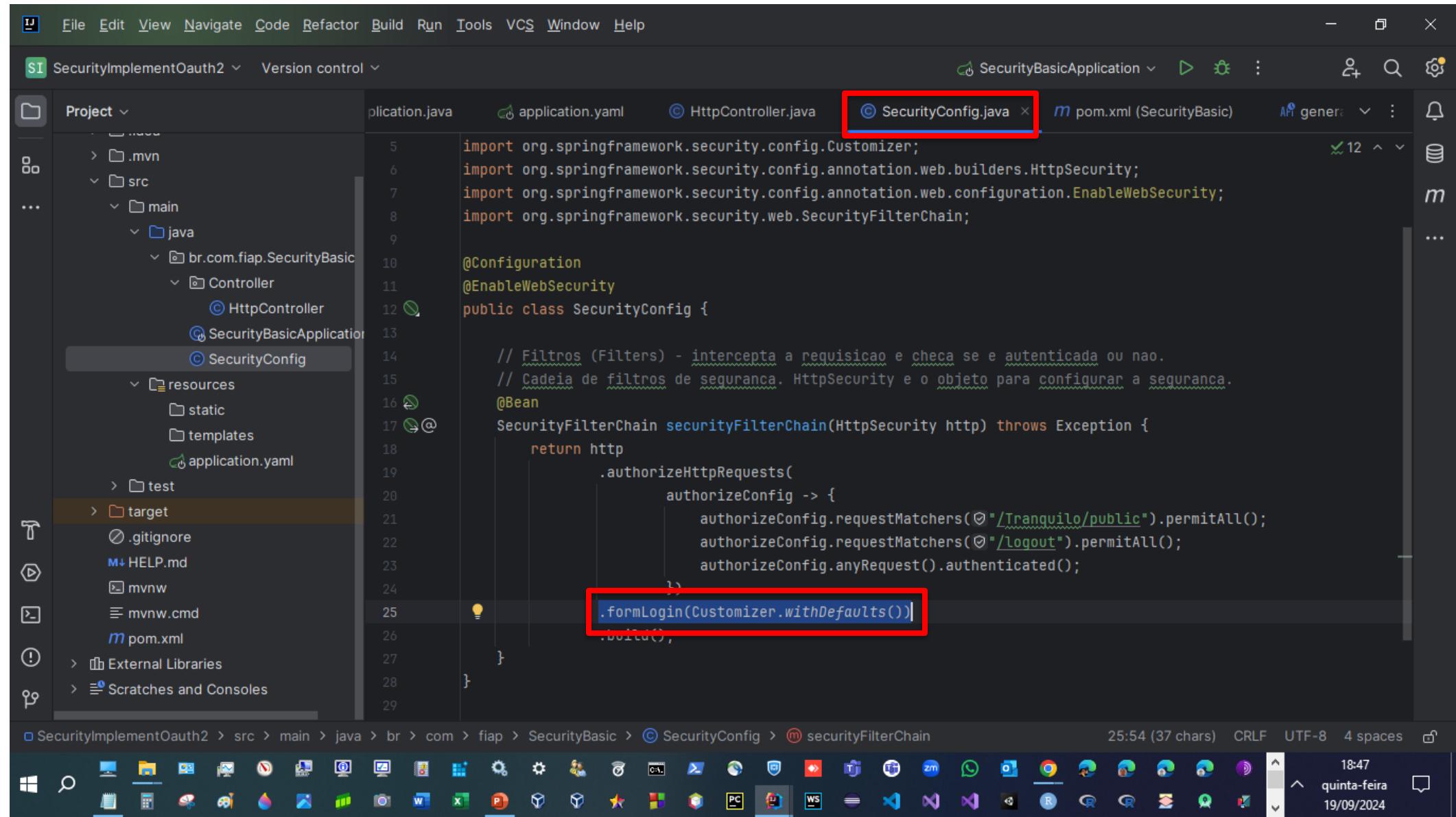
The screenshot shows the Google Cloud Platform (GCP) API Credentials interface. The left sidebar is collapsed, and the main area displays the 'Credenciais' (Credentials) section under the 'APIs e serviços' (APIs and services) heading. A red box highlights the 'IDs do cliente OAuth 2.0' (OAuth 2.0 Client IDs) table, which lists a single entry: 'teste-spring-security' (Type: Aplicativo da Web, ID do cliente: 724180968308-npmo...). Below this, another red box highlights a success message box that says 'Cliente OAuth criado' (OAuth client created). The bottom status bar shows the date and time as '18:44 quinta-feira 19/09/2024'.

| Nome | Data da criação | Tipo | ID do cliente | Ações |
|-----------------------|--------------------|-------------------|----------------------|-------|
| teste-spring-security | 19 de set. de 2024 | Aplicativo da Web | 724180968308-npmo... | |

Cliente OAuth criado

The screenshot shows a Java IDE interface with the following details:

- File Bar:** File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
- Title Bar:** SI SecurityImplementOauth2 Version control SecurityBasicApplication
- Project Explorer:** Shows the project structure with files like .mvn, src/main/java, and application.yaml.
- Code Editor:** The application.yaml file is open, showing Spring Security OAuth2 configuration. A red box highlights the client-id value: `client-id: 724180968308-npmodk06cuer0ojnlbmq3kqalbuk7uj.apps.googleusercontent.com`.
- Bottom Status Bar:** Shows the path SecurityImplementOauth2 > src > main > resources > application.yaml, encoding (9:1 LF), character set (ISO-8859-1), and spaces (2 spaces).
- Taskbar:** Shows various application icons including Windows, Task View, File Explorer, and several browser and utility icons.
- System Tray:** Shows the date and time (18:52, quinta-feira, 19/09/2024).



File Edit View Navigate Code Refactor Build Run Tools VCS Window Help

SI SecurityImplementOauth2 Version control SecurityBasicApplication pom.xml (SecurityBasic) API gener... : 12 ... m ...

Project

src/main/java
br.com.fiap.SecurityBasic
Controller
HttpController
SecurityBasicApplication
SecurityConfig
resources
static
templates
application.yaml
test
target
.gitignore
HELP.md
mvnw
mvnw.cmd
pom.xml
External Libraries
Scratches and Consoles

application.java application.yaml HttpController.java SecurityConfig.java pom.xml (SecurityBasic)

```
import org.springframework.security.config.Customizer;
import org.springframework.security.config.annotation.web.builders.HttpSecurity;
import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
import org.springframework.security.web.SecurityFilterChain;

@Configuration
@EnableWebSecurity
public class SecurityConfig {

    // Filtros (Filters) - intercepta a requisicao e checa se e autenticada ou nao.
    // Cadeia de filtros de segurança. HttpSecurity e o objeto para configurar a segurança.
    @Bean
    SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {
        return http
            .authorizeHttpRequests()
            .authorizeConfig -> {
                authorizeConfig.requestMatchers("/Tranquilo/public").permitAll();
                authorizeConfig.requestMatchers("/logout").permitAll();
                authorizeConfig.anyRequest().authenticated();
            }
            .formLogin(Customizer.withDefaults())
            .build();
    }
}
```

25:54 (37 chars) CRLF UTF-8 4 spaces

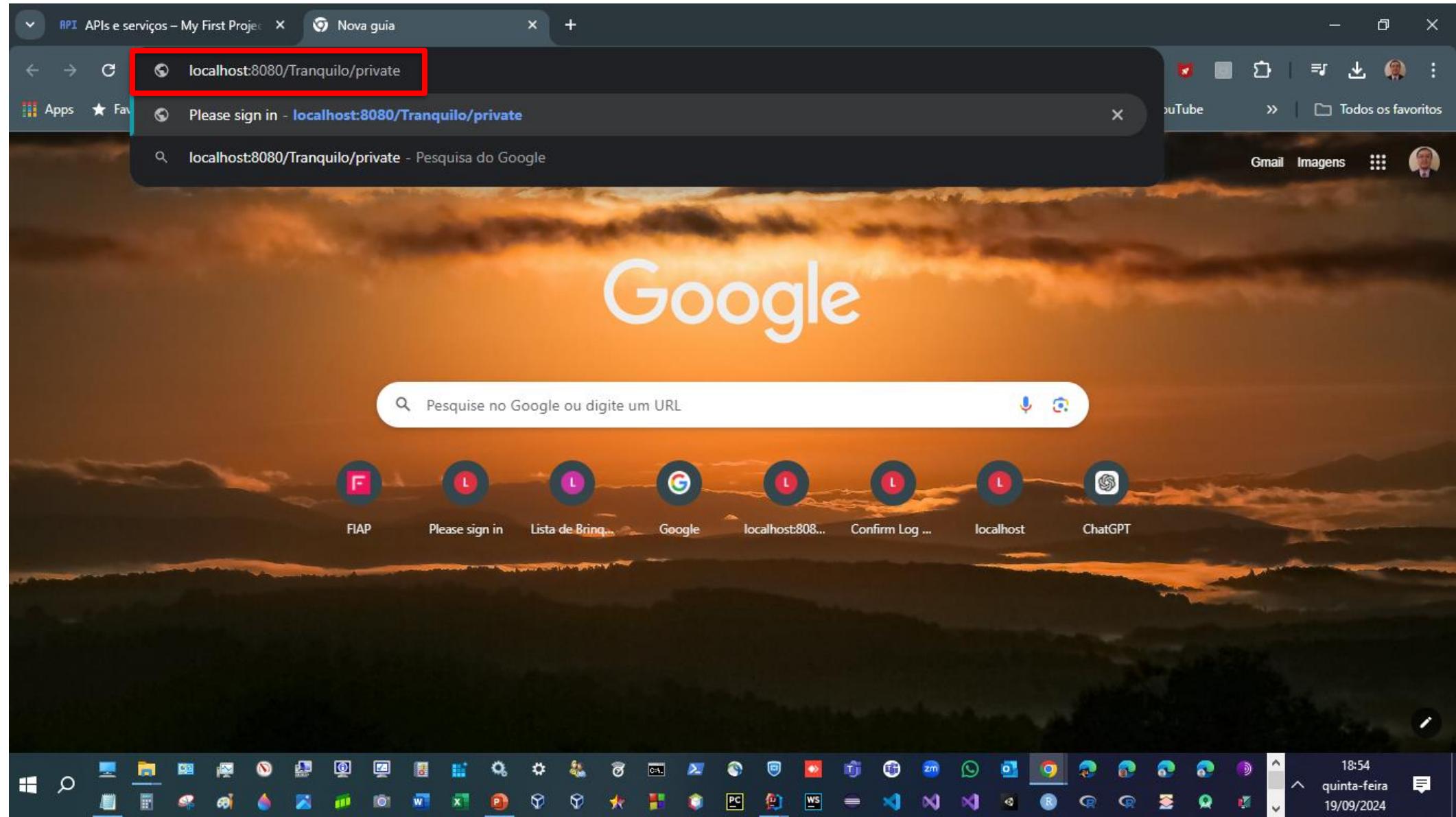
18:47 quinta-feira 19/09/2024

The screenshot shows a Java code editor in an IDE. The project structure on the left includes a .idea folder, a .mvn folder, and a src folder containing main and test directories. The main directory has java, resources, static, templates, and application.yaml files. The test directory contains target, .gitignore, HELP.md, mvnw, mvnw.cmd, and pom.xml. The pom.xml file is currently selected in the project tree. The code editor window shows SecurityConfig.java, which is annotated with @Configuration and @EnableWebSecurity. It defines a @Bean named securityFilterChain that returns an HttpSecurity object. The configuration includes authorizeHttpRequests, authorizeConfig, and anyRequest().authenticatedRoute(). A call to .oauth2Login(Customizer.withDefaults()) is highlighted with a red box. The status bar at the bottom shows the file path as SecurityImplementOauth2 > src > main > java > br > com > fiap > SecurityBasic > SecurityConfig > securityFilterChain, and the current time as 25:56 (39 chars) CRLF UTF-8 4 spaces.

```
import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;
import org.springframework.security.web.SecurityFilterChain;

@Configuration
@EnableWebSecurity
public class SecurityConfig {

    // Filtros (Filters) - intercepta a requisicao e checa se e autenticada ou nao.
    // Cadeia de filtros de segurança. HttpSecurity e o objeto para configurar a segurança.
    @Bean
    SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {
        return http
            .authorizeHttpRequests(
                authorizeConfig -> {
                    authorizeConfig.requestMatchers(ignored).permitAll();
                    authorizeConfig.requestMatchers("/Logout").permitAll();
                    authorizeConfig.anyRequest().authenticated();
                })
            .oauth2Login(Customizer.withDefaults())
            .build();
    }
}
```



Fazer login com o Google

Escolha uma conta

para prosseguir para [Aplicativo da Web](#)

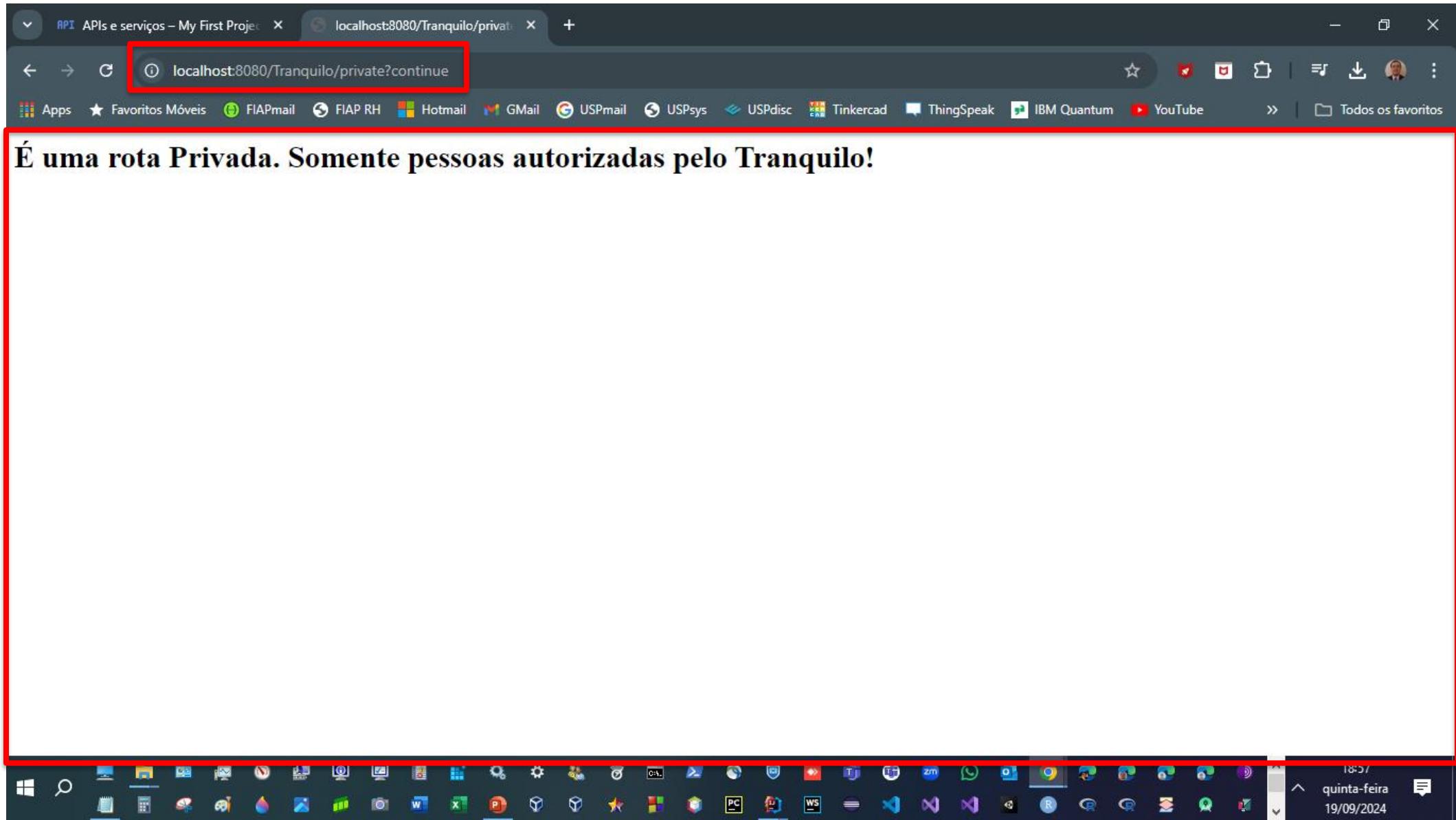
[Usar outra conta](#)

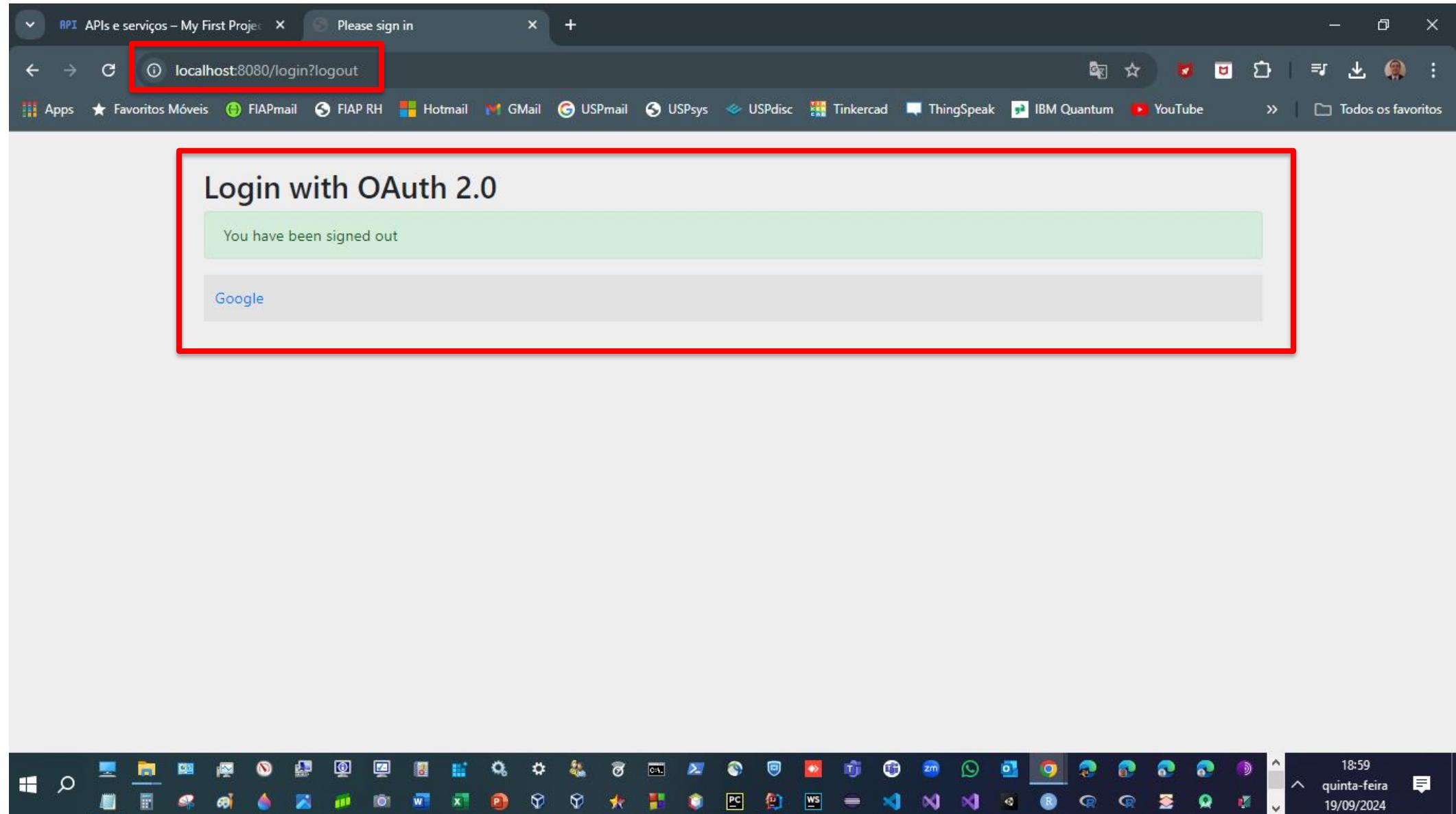
Português (Brasil) ▾

Ajuda Privacidade Termos

18:55 quinta-feira 19/09/2024

A screenshot of a Microsoft Edge browser window. The address bar shows a Google OAuth URL. The main content is a Google sign-in dialog titled 'Fazer login no serviço Aplicativo da Web'. It features a user profile placeholder, a 'Cancelar' button, and a 'Continuar' button. A red box highlights the 'Continuar' button. At the bottom, there's a language dropdown set to 'Português (Brasil)', and links for 'Ajuda', 'Privacidade', and 'Termos'. The taskbar at the bottom is visible, showing various pinned icons.





Referências para a Apresentação

Referências

George Coulouris, Jean Dollimore, Tim Kindberg, and Gordon Blair. **Sistemas Distribuídos: Conceitos e Projeto**. Bookman Editora, 5 edition, 2013.

Harvey M Deitel, Paul J Deitel, David R Choffnes, et al. **Sistemas Operacionais**. Pearson/Prentice Hall, 3 edition, 2005.

Maarten Van Steen and A Tanenbaum. **Sistemas Distribuídos: Princípios e Paradigmas**. Pearson/Prentice Hall, 2 edition, 2007.

Harvey M Deitel and Paul J Deitel. **Java, como programar**. Ed. Pearson/Prentice Hall, 8 edition, 2010.

StackOverflow. Disponível em: <<https://pt.stackoverflow.com/>>. Acesso em: abril de 2023.

GAMMA, Erich et al. **Elements of Reusable Object-Oriented Software**. Design Patterns, 1995.

COOPER, James William. **Java design patterns: a tutorial**. 2000.

GUERRA, Eduardo. **Design Patterns com Java: Projeto orientado a objetos guiado por padrões**. Editora Casa do Código, 2014.

Referências para a Apresentação

Referências

AWS Amazon – O que é RESTful. Disponível em: <[https://aws.amazon.com/pt/what-is/restful-api/#:~:text=Representational%20State%20Transfer%20\(REST\)%20is,complex%20network%20like%20the%20internet,>](https://aws.amazon.com/pt/what-is/restful-api/#:~:text=Representational%20State%20Transfer%20(REST)%20is,complex%20network%20like%20the%20internet,>)>. Acesso em: maio de 2023.

InfoQ. Disponível em: <<https://www.infoq.com/minibooks/emag-03-2010-rest>>. Acesso em: maio de 2024.

Michelli Brito. Disponível em: <<https://www.youtube.com/@MichelliBrito>>. Acesso em: abril de 2024.

Obeautifulcode. Disponível em: <<http://blog.obeautifulcode.com/API/Learn-REST-In-18-Slides/>>. Acesso em: junho de 2022.

Berkeley. Disponível em: <<http://courses.ischool.berkeley.edu/i290-rmm/s12/slides/Lecture3%20REST.pdf>>. Acesso em: maio de 2023.

Trybe. Disponível em: <<https://blog.betrybe.com/tecnologia/deploy/>>. Acesso em: julho de 2024.

Obrigado!

Contato: profmarcel.wagner@fiap.com.br

Cursos:

Tecnologia em Análise e Desenvolvimento de Sistemas (TDS)

Tecnologia em Defesa Cibernética (TDC)

Engenharia de Software (ES)

