

FIAP

FACULDADE DE INFORMATICA E ADMINISTRAÇÃO PAULISTA

## TECH CHALLENGE – BLOGGING DINAMICO OUTSYSTEMS

### PARTICIPANTES:

Dilson Ferreira Santana Rm366948

Emerson Chalegre Rm 366946

Marcelo Rm 366958

Germano Leite Rm 366947

João Luís Rm366954

Turma : Pos Tech - Dev Foudantions - BB

BELO HORIZONTE- MG- 2026

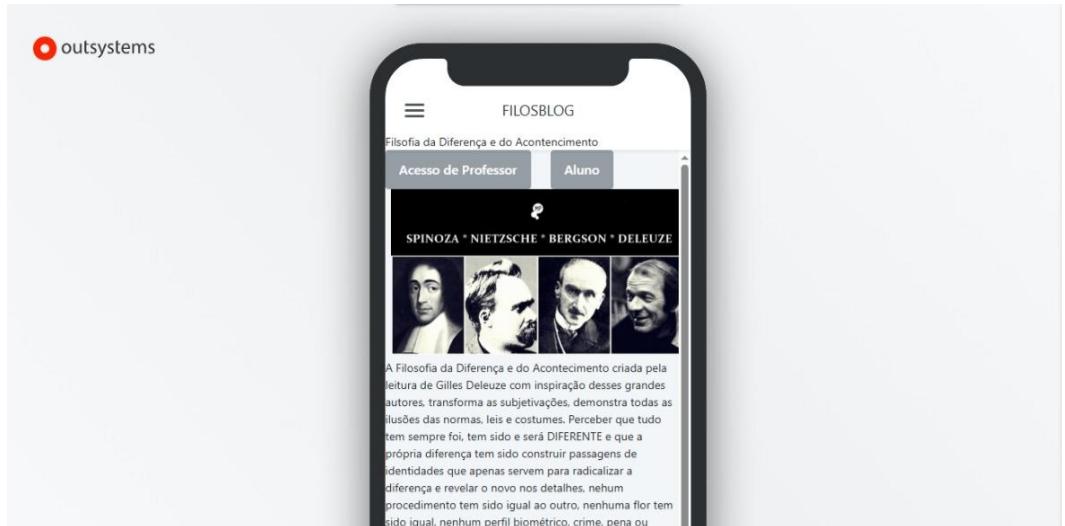
## **INTRODUÇÃO**

Trabalho realizado para desenvolvimento de uma aplicação em Low Code da Fase 4 – Low Code e Mobile. Por meio dos conhecimentos adquiridos utilizando a plataforma OutSystems. A aplicação deve permitir que alunos e alunas visualizem uma lista de posts, leiam postagens específicas e ofereça uma visão administrativa para os (as) docentes, permitindo a criação, edição, listagem e exclusão de postagens.

Sendo assim focamos na separação de permissões e a gestão de conteúdo.

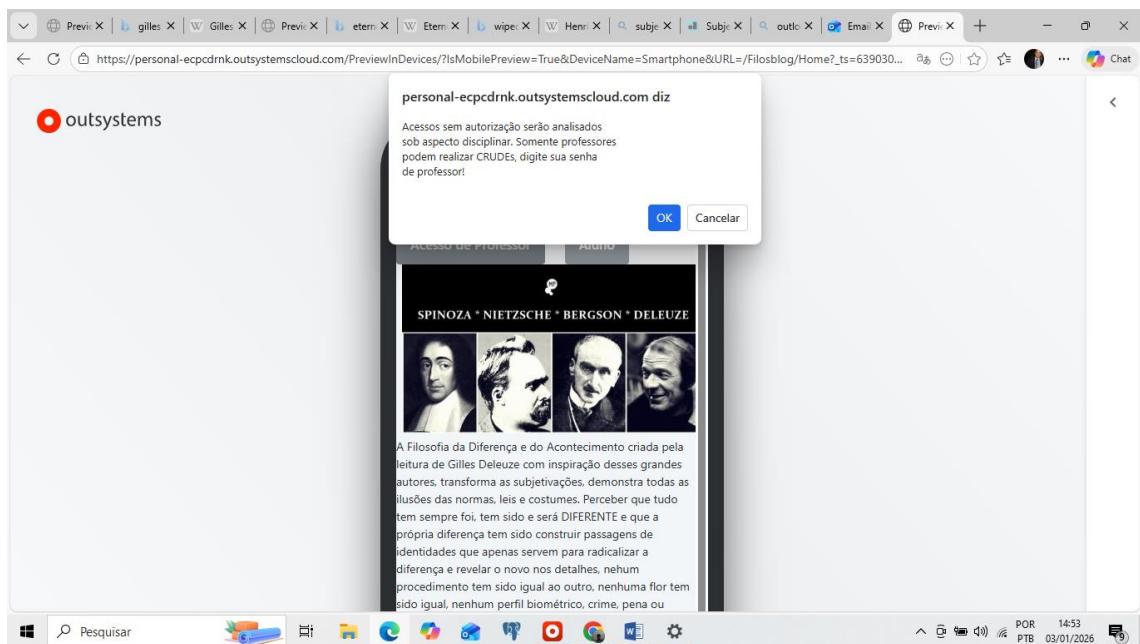
O processo de trabalho foi realizando utilizando as aulas sobre Outsystems como auxiliar na ambientação com a nova plataforma. Tivemos dificuldade em algumas separações de permissões, demoramos encontrar o link que permitia CRUDEs estava dentro de uma lista e somente depois de muitas tentativas conseguimos isola-la para permitir que somente professores realizassem funções de administrador, essa dificuldade surgiu pelo motivo da plataforma auto completar as edições iniciais, ou seja, a facilidade de evitar digitar o código linha por linha também traz dificuldades para encontrar formas de alterar o projeto.

# DESENVOLVIMENTO



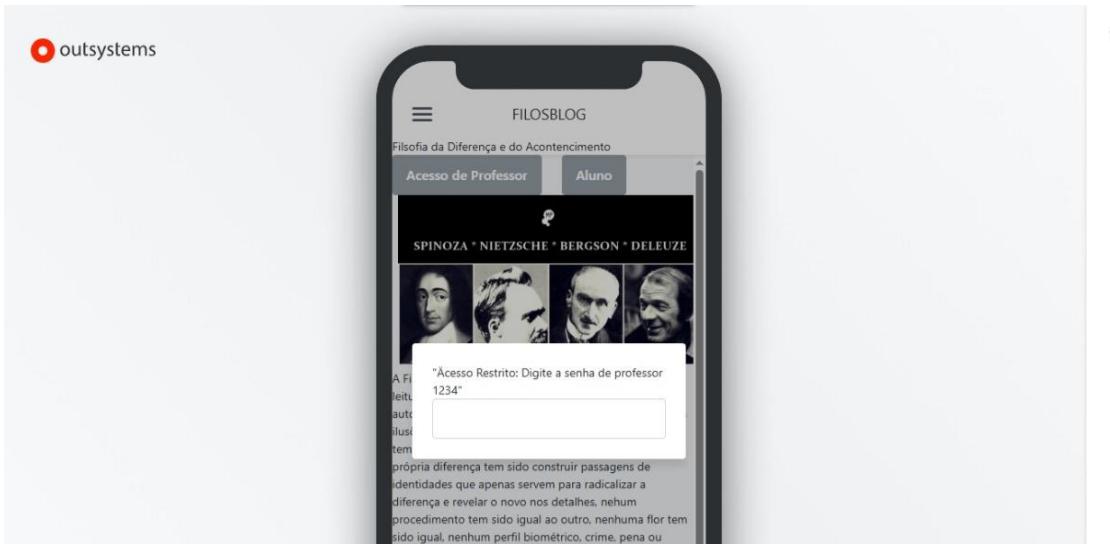
1

Figura 1 - Tela Inicial com os botões “Acesso de Professor” e “Aluno”



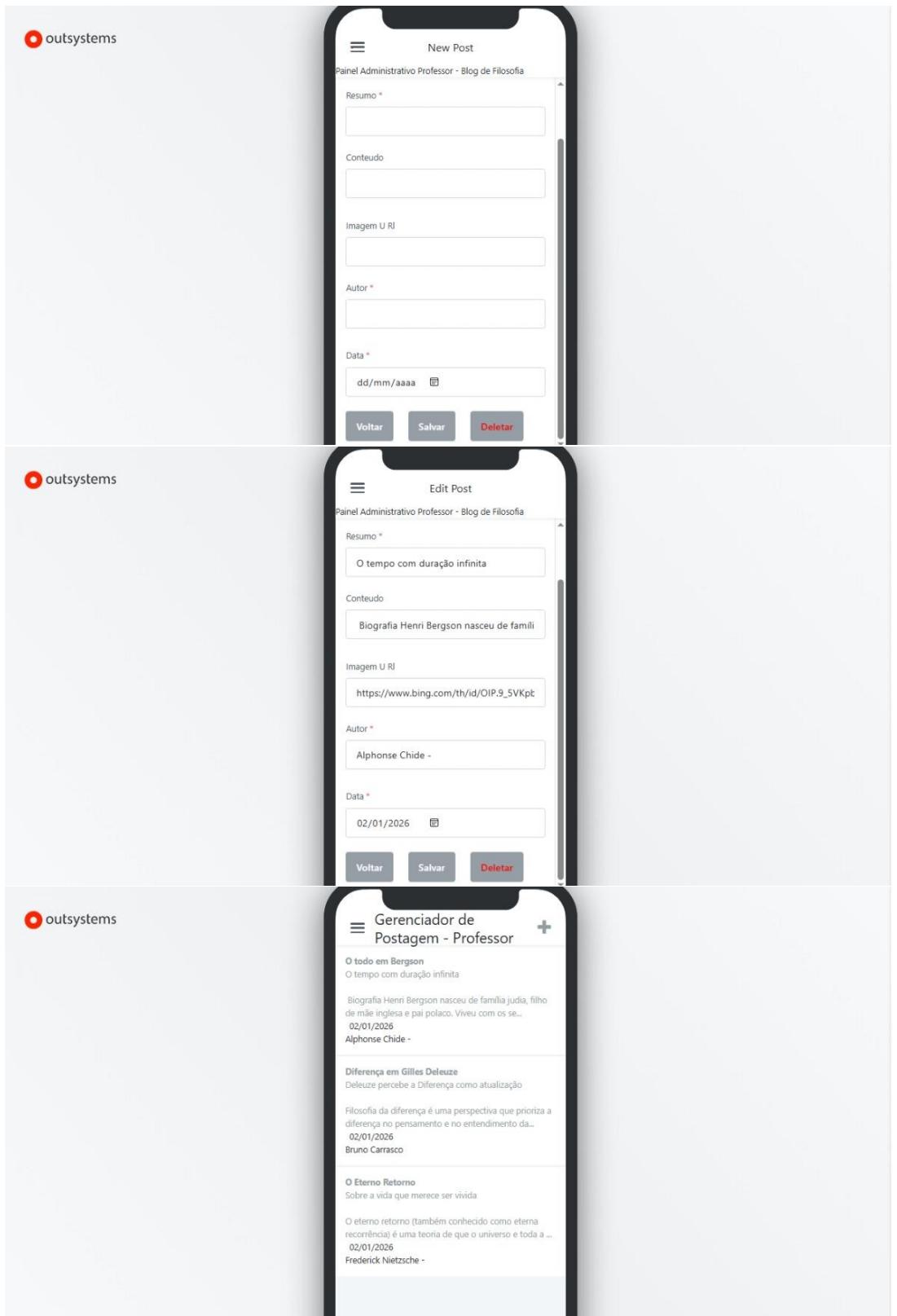
2

Figura 2 – Tela demonstração de separação de ambientes Administrador x Cliente com aviso



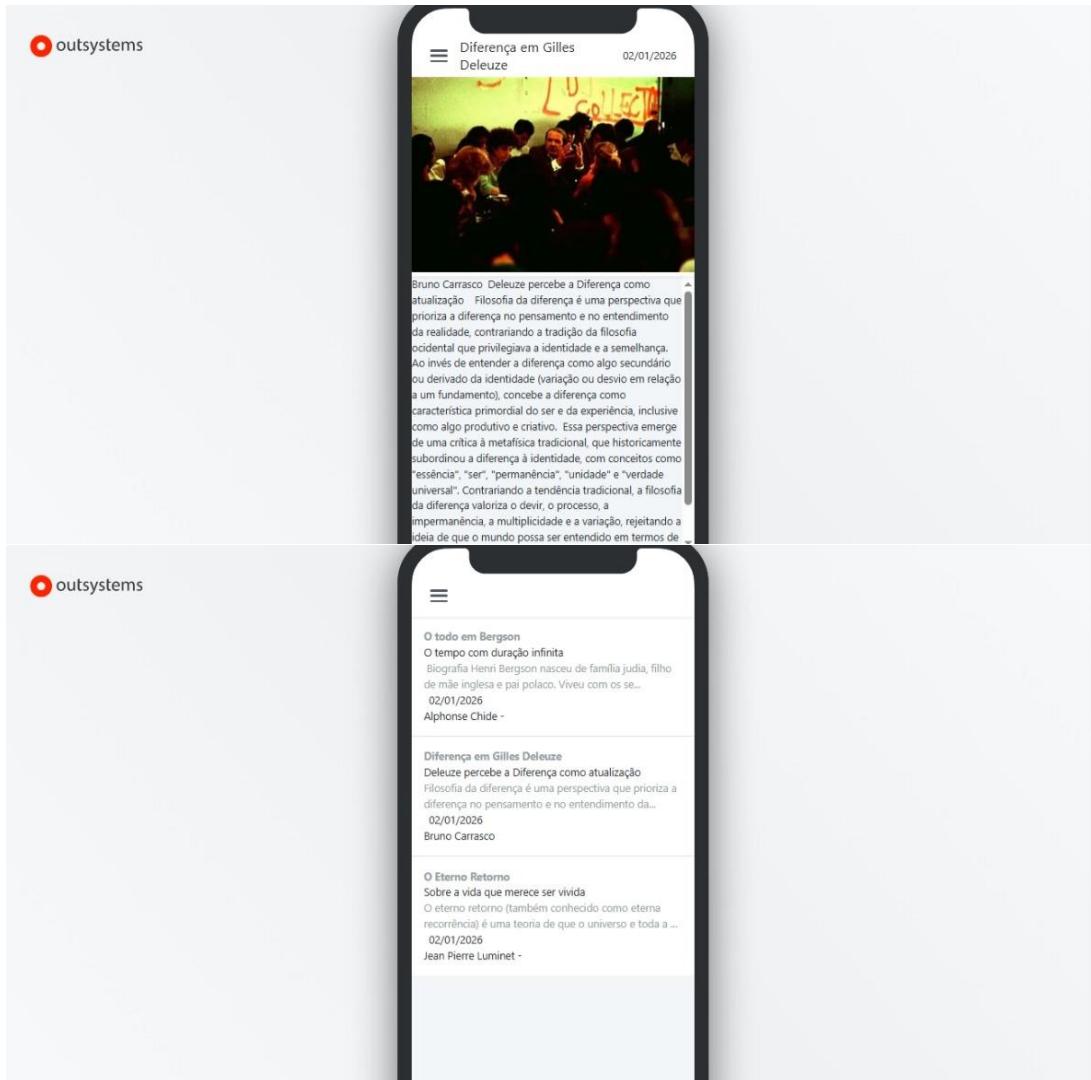
3

Figura 3 – Tela de login, tem vista tratar-se de um blogging optamos por login apenas para o perfil de Professor administrador e acesso livre para leitura de alun@s.



4

Figura 4 – Telas com visão professor possibilitando CRUD, o ícone “+” abre a função de criar, Professor lê o post clicando no título e no conteúdo e edita clicando no resumo.



5

Figura 5 – Telas da visualização de Aluno que tem link apenas no título e no conteúdo e somente abre para leitura

**Figure 6 - Telas demonstrando a ação do popup do login se clicarmos no “Acesso de professor”, simulando autenticação que foi descartada que num ambiente fechado seria controlado por roles.**

The figure consists of two screenshots of the OutSystems Service Studio interface, illustrating the development and deployment of a login feature for a "Filosblog" application.

**Screenshot 1: Logic Flow (MainFlow > Home > Input\_loginuserMeuOnChange)**

This screenshot shows the logic flow diagram. It starts with a "Start" node, followed by a decision diamond labeled "loginuserMeu = 1234?". If the condition is "True", it leads to a "MainFlow/Posts" action and then an "End" node. If the condition is "False", it leads directly to the "End" node. The logic is controlled by the variable "loginuserMeu".

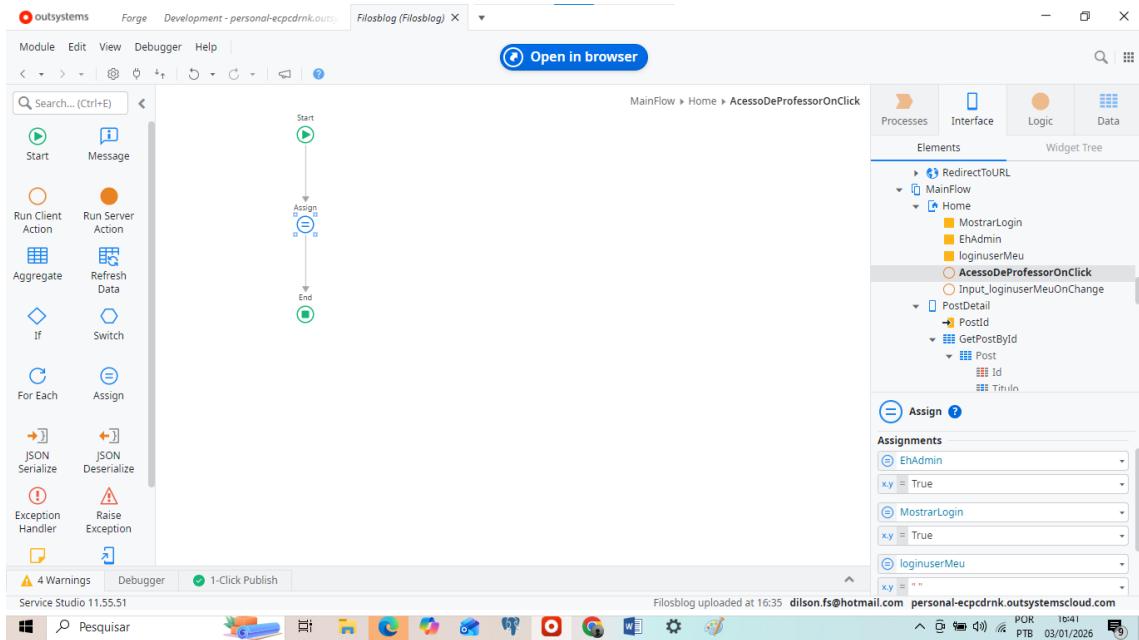
**Screenshot 2: User Interface (Filosblog Home Screen)**

This screenshot shows the user interface of the Filosblog application. The main screen displays a header with "FILOSLOG" and navigation links like "Filosofia da Diferença e do Acontecimento", "Acesso de Professor", and "Aluno". Below the header is a banner featuring portraits of Spinoza, Nietzsche, Bergson, and Deleuze. A modal dialog box is displayed, prompting the user to enter a password: "Acesso Restrito: Digite a senha de professor 1234".

**Widget Tree and Details**

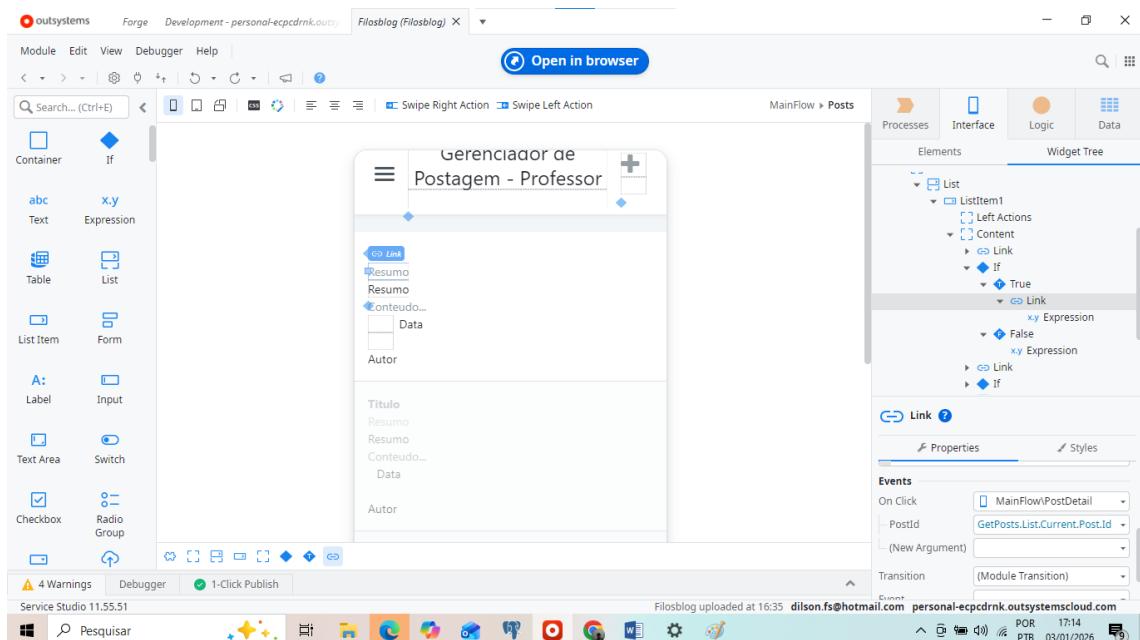
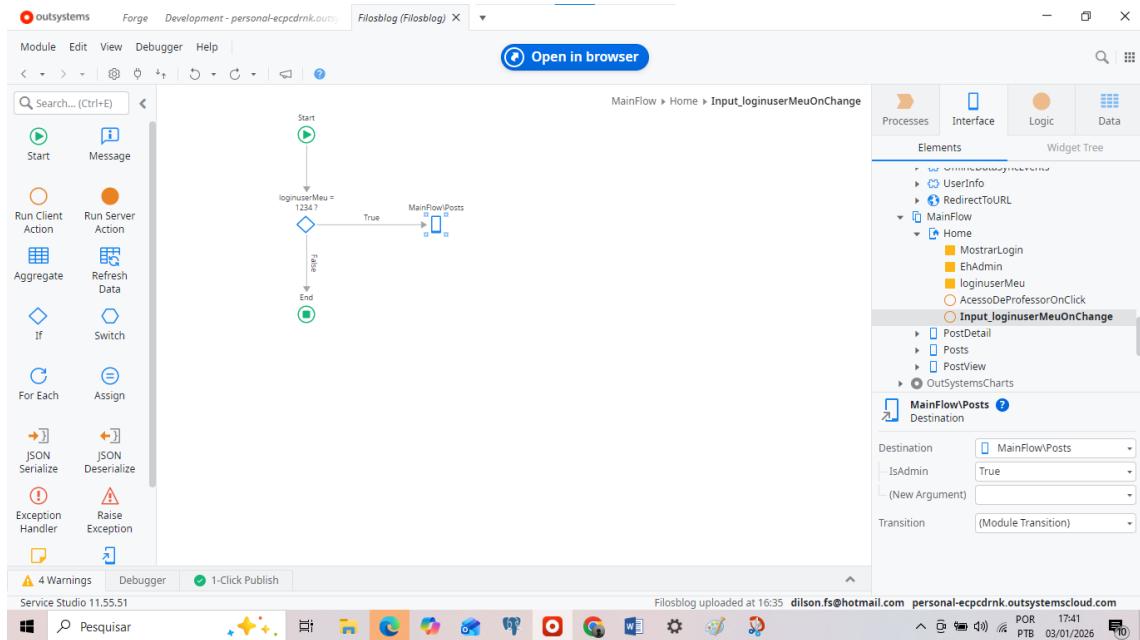
The right side of both screenshots shows the "Widget Tree" and "Details" panels. The "Widget Tree" shows the structure of the screen, including sections like "HeaderLeft", "Title", "HeaderRight", "HeaderContent", and "Content". The "Details" panel provides specific information for selected elements, such as the "Input\_loginuserMeu" field, which is described as a "Client Action" with a name of "Input\_loginuserMeuOnChange".

6



7

Figura 7 – Tela da ação de escolher para quem mostrar o login e limpar a senha para evitar o cache, optamos por deixar o acesso de alun@ sem senha e sem login para contemplar o intuito do blogging.



8

Figura 8 – Árvore do If que demonstra o link para a tela PostDetail onde fazemos CRUDE abre ao clicar no Resumo, se for verdade que clicou e passou pelo input da senha de professor abre o link para criar, alterar, salvar/atualizar e deletar. Nesta tela repare que temos dois “resumos” um com link e outro sem o If encaminha um para o profess@r e outro para o alun@.

The screenshot displays two main windows from the OutSystems Service Studio:

- MainFlow UI Structure:** This window shows the navigation flow. It starts at the **Home** screen, which has a link to **Posts**. From **Posts**, there is a link to **PostView**. On the left, there is a toolbar with icons for **Screen**, **Block**, **Email**, **External Site**, and **Comment**. The right side features a tree view of the MainFlow elements, including Home, Posts, PostView, and various sub-components like OutSystemsCharts, OutSystemsMaps, and OutSystemsUI.
- Post Data Database:** This window shows a table named **Post** with columns **Id**, **Título**, and **Resumo**. The data includes three rows:
 

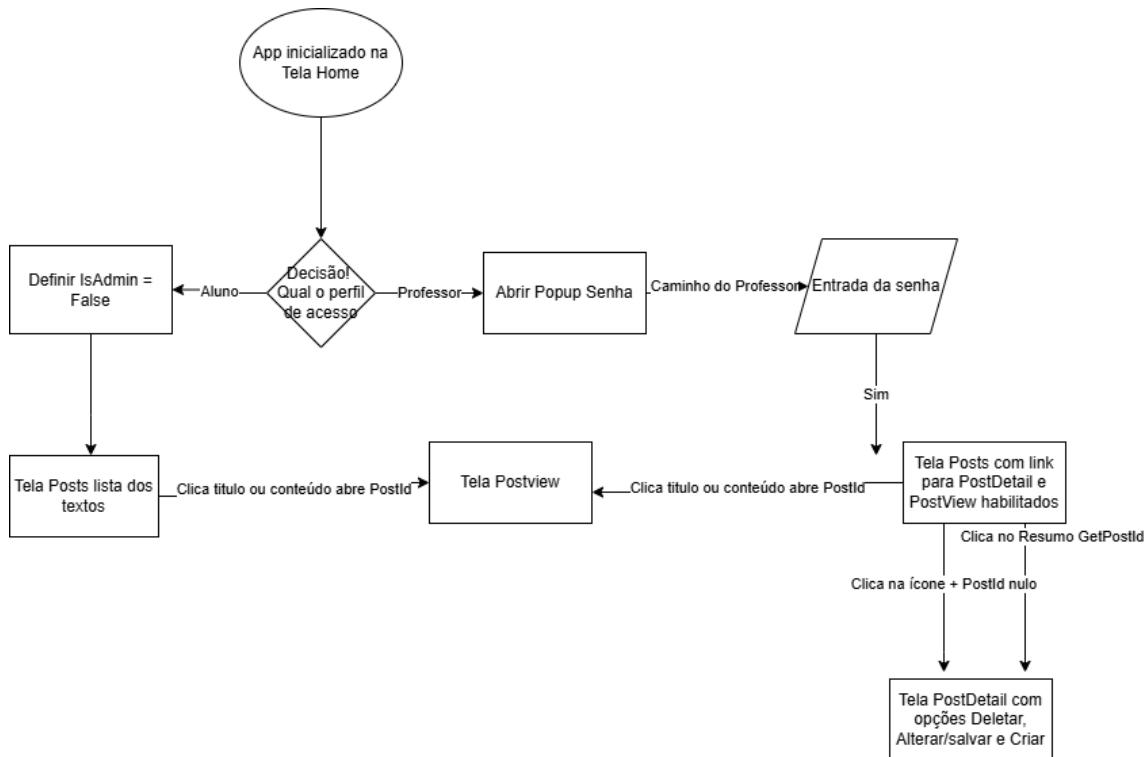
Id	Título	Resumo
9	O Eterno Retorno	Sobre a vida que merece ser vivida
8	Diferença em Gilles Deleuze	Deleuze percebe a Diferença como atualização
2	O todo em Bergson	O tempo com duração infinita Henri Bergson (Paris, 18 de outubro de 1859 — Paris, 4 de janeiro de 1941) foi um filósofo e diplomata francês. Ele é considerado um dos principais representantes do filosofia francesa contemporânea. Sua obra mais famosa é "O todo em Bergson", publicada em 1907, na qual ele defende a teoria da "duração" como a base da realidade.

 The right side of this window shows the database schema with entities like **FilosblogDataModel** and **Post**, which contains attributes **Id**, **Título**, **Resumo**, **Conteúdo**, **Autor**, **Data**, and **ImagemURI**, along with methods for creating, updating, and deleting posts.

9

Figura 9 - Por fim as telas da estrutura de todas as telas e o Database. A navegação inicia na tela Home onde o usuário escolhe seu perfil Administrad@r (profess@r) ou Cliente(alun@), dessa para a tela Posts onde permite a visualização da lista de textos, depois o sistema já guardando o perfil oferece a escolhe de leitura de texto em específico clicando no título ou no conteúdo e no caso do docente pode clicar no resumo para editar ou no ícone + para criar novo post.

## Fluxograma



10 Figura 10 – Fluxograma de navegação e estados.

O fluxograma demonstra a transição de estados da aplicação, onde a navegação para a PostDetail pode ocorrer via seleção de Id existente (edição) ou via NullIdentifier (Novo Post), dependendo da ação do usuário@ no cabeçalho ou no corpo da solução.

[Link para o repositório do código-fonte no Github](#)

Repositório:

<https://github.com/Dilsonai/TC-fase-4-OutSystems->

Visibilidade: Público

### Orientações de Utilização da Aplicação

Para testar a aplicação de forma imediata, utilize o link de visualização em nuvem: Link da sua aplicação OutSystems.

### Passo a passo para teste:

- Acesso Inicial:** Na tela "Home", selecione o perfil desejado.
- Perfil Alun@:** Navegue pela lista de posts e clique em um título para ler o conteúdo completo.
- Perfil Profess@r:** Insira a credencial "1234". Note que o ícone "+" e os links de edição serão habilitados nas telas "Posts" e "PostDetail".
- Persistência:** O sistema mantém o perfil escolhido durante toda a navegação através de parâmetros de entrada.

## **Link para a Apresentação Gravada**

O vídeo abaixo demonstra o funcionamento da aplicação, as regras de negócio aplicadas e a diferenciação entre os perfis de acesso (Profess@r e Alun@).

- **Link do Vídeo:** <https://youtu.be/LgkfcImzdZo>