

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA
FACULDADE DE TECNOLOGIA DE BOTUCATU
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO
DE
SISTEMAS**

VINÍCIUS DADÁRIO DE FREITAS

APLICATIVO DE GERENCIAMENTO DE ANÚNCIOS LOCAIS

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA
FACULDADE DE TECNOLOGIA DE BOTUCATU
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO
DE
SISTEMAS**

VINICIUS DADÁRIO DE FREITAS

APLICATIVO DE GERENCIAMENTO DE ANÚNCIOS LOCAIS

Orientador: Prof. Dr. Osvaldo Cesar Pinheiro de Almeida

Artigo entregue como Trabalho de Conclusão de Curso apresentado à FATEC - Faculdade de Tecnologia de Botucatu, para obtenção do título de Tecnólogo no Curso Superior de Análise e Desenvolvimento de Sistemas.

Botucatu - SP
Novembro - 2024

APLICATIVO DE GERENCIAMENTO DE ANÚNCIOS LOCAIS

LOCAL ADVERTISEMENT MANAGEMENT APPLICATION

Vinicius Dadário de Freitas¹ Osvaldo Cesar Pinheiro de Almeida²

¹ ReGraduando em Análise e Desenvolvimento de Sistema, e Faculdade de Tecnologia de Botucatu, Prudente de Moraes, 697, 18606-060, Botucatu, São Paulo email: viniciusddrft5@gmail.com.

² Professor Doutor da Faculdade de Tecnologia de Botucatu, Avenida José Ítalo Bacchi s/n – Jardim Aeroporto – Botucatu – SP CEP.18606-855 – osvaldo.almeida@fatec.sp.gov.br.

RESUMO

Este artigo explora o desenvolvimento de um aplicativo móvel para gestão remota de sinalização digital, uma tecnologia que transforma a comunicação estática em uma experiência dinâmica e interativa por meio de telas eletrônicas. A sinalização digital tem se destacado como uma ferramenta essencial em locais públicos, como shoppings, aeroportos e hospitais, permitindo a exibição de conteúdos que variam de anúncios a informações importantes. O aplicativo desenvolvido busca simplificar o controle e a atualização desses conteúdos em tempo real, eliminando a necessidade de intervenção local. Um dos principais desafios enfrentados foi lidar com as restrições de acesso a um servidor web exclusivo de intranet, necessário para garantir a segurança do sistema. Para superar essa barreira, o Firebase foi escolhido como intermediário, possibilitando o armazenamento e a recuperação dos dados de forma eficiente e segura. Essa abordagem resultou em uma solução robusta e escalável, capaz de atender às demandas modernas de gerenciamento de sinalização digital e melhorar a experiência de uso.

Palavras-chave: Flutter. Dart. Aplicativo móvel. Firebase. Sinalização Digital.

ABSTRACT

This article explores the development of a mobile application for remote management of digital signage, a technology that transforms static communication into a dynamic and interactive experience through electronic displays. Digital signage has emerged as an essential tool in public spaces such as shopping malls, airports, and hospitals, enabling the display of a wide range of content, from advertisements to important information. The developed application aims to simplify the control and real-time updating of this content, eliminating the need for on-site intervention. One of the main challenges faced was addressing access restrictions to a web server operating exclusively on an intranet, required to ensure system security. To overcome this barrier, Firebase was chosen as an intermediary, enabling efficient and secure data storage and retrieval. This approach resulted in a robust and scalable solution capable of meeting modern digital signage management demands and enhancing the user experience.

Key Words: Flutter. Dart. Mobile application. Firebase. Digital Signage.

1 INTRODUÇÃO

Antes de mergulharmos no cerne deste artigo, é fundamental compreender o conceito central que impulsiona esta investigação: a digital signage, ou sinalização digital. Esta tecnologia vai além da abordagem tradicional e estática da publicidade, transformando-a em uma forma dinâmica, interativa e altamente personalizável. Por meio de telas eletrônicas estrategicamente posicionadas em espaços públicos, a sinalização digital exibe uma gama de conteúdos, que podem variar desde anúncios publicitários e informações relevantes até elementos visuais que enriquecem o ambiente ao nosso redor. Combinando design, tecnologia e comunicação, a sinalização digital cria uma sinergia entre os espaços físicos e o mundo digital, oferecendo uma experiência que se adapta, cativa e informa (BAUER, 2011).

No contexto atual, em que a informação e a comunicação desempenham um papel central em nossa vida cotidiana, a sinalização digital emergiu como uma poderosa ferramenta de interação e engajamento nos espaços públicos. Ela ultrapassa as barreiras da comunicação estática, abrindo novas possibilidades para transmitir mensagens e conteúdos de maneira dinâmica e impactante. Desde a exibição vibrante de anúncios em shoppings movimentados até a apresentação de informações cruciais em aeroportos e hospitais, a sinalização digital se consolidou como um componente essencial na paisagem urbana moderna.

Este artigo descreve o desenvolvimento de um aplicativo móvel projetado para otimizar e simplificar a gestão de sinalização digital. O principal objetivo do aplicativo é oferecer aos usuários a capacidade de controlar e atualizar conteúdo em tempo real, de forma remota, eliminando a necessidade de presença física junto às telas de exibição. No entanto, um dos desafios significativos enfrentados durante o desenvolvimento foi a criação de uma solução que respeitasse as restrições de acesso ao servidor web onde os anúncios são exibidos. Este servidor deveria operar exclusivamente em uma intranet para garantir a segurança, e o aplicativo móvel, por sua vez, não poderia acessar diretamente esse servidor.

Para superar essa barreira, optou-se pelo uso do Firebase como intermediário, permitindo que o aplicativo se comunicasse de forma segura e eficiente com o servidor web sem violar as regras de acesso. O Firebase possibilitou que os anúncios fossem armazenados e recuperados de maneira segura, facilitando a integração entre o aplicativo e o servidor de exibição. Dessa forma, o Firebase não apenas cumpriu a função de intermediário, mas também contribuiu para a construção de uma arquitetura robusta e escalável, essencial para garantir a eficiência do sistema.

Ao longo deste artigo, exploraremos a concepção e o desenvolvimento dessa solução, detalhando as tecnologias adotadas, os desafios enfrentados e os resultados alcançados.

2 MATERIAL E MÉTODOS

O desenvolvimento deste projeto teve início com a escolha da linguagem Dart, projetada para a criação de interfaces de usuário de alta performance. Desenvolvida pelo Google, ela foi selecionada por sua integração com o Flutter, o que permitiu o desenvolvimento eficiente de aplicações multiplataforma. A linguagem foi utilizada como base para a criação da versão web do sistema, garantindo robustez e escalabilidade na comunicação com o backend e no processamento das funcionalidades (Dart, 2024).

O Flutter, framework multiplataforma também desenvolvido pelo Google, foi escolhido para o desenvolvimento das versões web e mobile do sistema. Um framework refere-se a uma estrutura de software que fornece ferramentas e funcionalidades pré-definidas para facilitar o desenvolvimento de aplicações, como bibliotecas reutilizáveis e padrões de arquitetura. Ele oferece uma arquitetura limpa que facilita a reutilização de código entre as plataformas, reduzindo custos e esforços de manutenção. Na aplicação web, o Flutter é responsável por recuperar anúncios armazenados no Firebase a cada 5 minutos e exibi-los em loop, respeitando o tempo de exibição configurado para cada anúncio. O Flutter proporcionou uma experiência uniforme em ambas as versões, mantendo a qualidade da interface e a eficiência do sistema (Flutter, 2024).

Para superar as limitações de acesso direto ao servidor web pelos dispositivos móveis, o Firebase foi utilizado como intermediário seguro e eficiente. O Firebase é uma plataforma desenvolvida pelo Google que oferece uma ampla gama de serviços para o desenvolvimento de aplicações web e mobile. Ele desempenha diversas funções essenciais no projeto: o Firestore foi usado para armazenar os anúncios de maneira organizada, permitindo que tanto o aplicativo quanto o servidor web acessem os dados de forma centralizada; o Authentication controla a criação de novos usuários por meio de tokens específicos, onde cada token – uma credencial digital que identifica um usuário ou aplicativo – gera uma hash SHA-512 que é comparada com a armazenada no Firebase para a autenticação e definição do tipo de usuário; o Storage guarda as imagens relacionadas aos anúncios, garantindo que estejam acessíveis para diferentes versões do sistema. O modelo de interação adotado assegura que o aplicativo mobile e o servidor web se comuniquem de forma eficiente e segura, sem violar as restrições de acesso impostas pelo ambiente intranet (Firebase, 2024).

O algoritmo SHA-512 (Secure Hash Algorithm 512) é uma função criptográfica de hash que transforma dados em uma sequência fixa de 512 bits. Ele é amplamente utilizado para garantir a integridade e segurança das informações, criando um identificador único e irreversível para os dados. No projeto, o SHA-512 é usado para gerar hashes de tokens, que são comparados aos armazenados no Firebase para validar a autenticidade e determinar o tipo

de usuário. A hash resultante da função garante que os dados sejam protegidos contra modificações, reforçando a segurança do sistema.

A versão web do sistema foi containerizada utilizando o Podman, uma solução leve e eficiente para gerenciamento de contêineres. O Podman permite criar, executar e gerenciar contêineres de forma eficiente e segura. Diferentemente de outras soluções, como o Docker, ele não exige a execução de um daemon, tornando-o mais leve e seguro. Essa abordagem simplifica o processo de implantação e manutenção do ambiente, permitindo configurar rapidamente o sistema e disponibilizá-lo para acesso nas TVs. Com o uso do Podman, o servidor web opera de forma isolada, garantindo maior segurança e facilidade na replicação do ambiente (Podman, 2024).

Durante o desenvolvimento, foi utilizado o Visual Studio Code, uma IDE conhecida por sua leveza e suporte robusto ao Dart e Flutter. Uma IDE (Integrated Development Environment) é um ambiente de desenvolvimento integrado que reúne ferramentas essenciais para a programação, como editor de código, depurador e gerenciador de projetos. O Visual Studio Code otimizou o fluxo de trabalho ao oferecer recursos como autocompletar, depuração integrada e uma ampla gama de extensões específicas para as tecnologias adotadas no projeto (Visual Studio Code, 2024).

Para a autenticação de tokens, foi implementado o algoritmo de hashing SHA-512. Quando um token é enviado, sua hash é gerada e comparada com a armazenada no Firebase, validando sua autenticidade e identificando o tipo de usuário.

Usuários comuns: possuem permissões para criar e excluir seus próprios anúncios, além de visualizar todos os anúncios disponíveis. Por padrão, os anúncios criados por usuários comuns têm duração de 30 segundos.

Administradores: têm permissões adicionais, como editar e excluir anúncios de qualquer usuário, além de ajustar a duração dos anúncios conforme necessário.

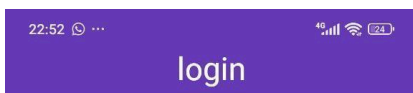
Essa estrutura de autenticação garante alta segurança e flexibilidade ao sistema, permitindo gerenciar usuários e anúncios de forma eficaz (GUERON, 2011).

3 RESULTADOS E DISCUSSÃO

3.1 Login

Na Figura 1 temos a tela de login que leva o usuário diretamente ao dashboard onde ele gerencia os anúncios, contém também um botão para levar a tela de cadastro de novos usuários, a imagem abaixo mostra essa tela nas versões em plataforma mobile.

Figura 1 – Login mobile

[Cadastrar-se](#)



Fonte: Próprio Autor, 2024.

Na Figura 2 existe a mesma tela de login porém em sua versão web, como todas funções reaproveitadas no dart/flutter e só mudando o layout para o tamanho adequado.

Figura 2 – Login web

Login

E-mail

Senha

Entrar

Cadastrar-se

Fonte: Próprio Autor, 2024.

3.2 Cadastro

Na Figura 3, Tela de cadastro de novos usuários, onde o token fornece a autorização necessária para a criação de um novo usuário, em seu layout adequado para a web.

Figura 3 - Cadastro web

Cadastro

E-mail

Senha

Token

Criar

Fonte: Próprio Autor, 2024.

Na Figura 4, Tela de cadastro de novos usuários, em seu layout adequado para a as plataformas mobile, com os exatos mesmos comportamentos.

Figura 4 - Cadastro mobile

The image shows a mobile registration screen titled "Cadastro". At the top, there is a purple header bar with a back arrow on the left and the word "Cadastro" in the center. Below the header, there are three input fields: "E-mail", "Senha" (with an eye icon for toggling visibility), and "Token". At the bottom of the form is a purple button labeled "Criar". The screen is displayed on a mobile device, with the status bar at the top showing the time 22:53, signal strength, Wi-Fi, and battery level (24%). The Android navigation bar is visible at the bottom.

Fonte: Próprio Autor, 2024.

3.3 Dashboard

Na Figura 6, Tela do dashboard para gerenciar anúncios, com a opção de apagar anúncios e um botão para acessar a tela de onboarding (disponível apenas no mobile), em seu layout para plataforma mobile.

Figura 6 - Dashboard mobile



Fonte: Próprio Autor, 2024.

Na Figura 7, Tela do dashboard onde é mostrado o que acontece ao gesto de clicar e segurar por um breve tempo em um anúncio, mostrando a opção de deletar o anúncio caso tenha as permissões necessárias.

Figura 7 - Demonstração de deletar anúncio mobile



Fonte: Próprio Autor, 2024.

Na Figura 8, Tela do dashboard onde é mostrado a tela de gerenciamento dos anúncios com a única diferença em funcionalidade da versão mobile sendo não ter o botão onde é possível adicionar um novo anúncio.

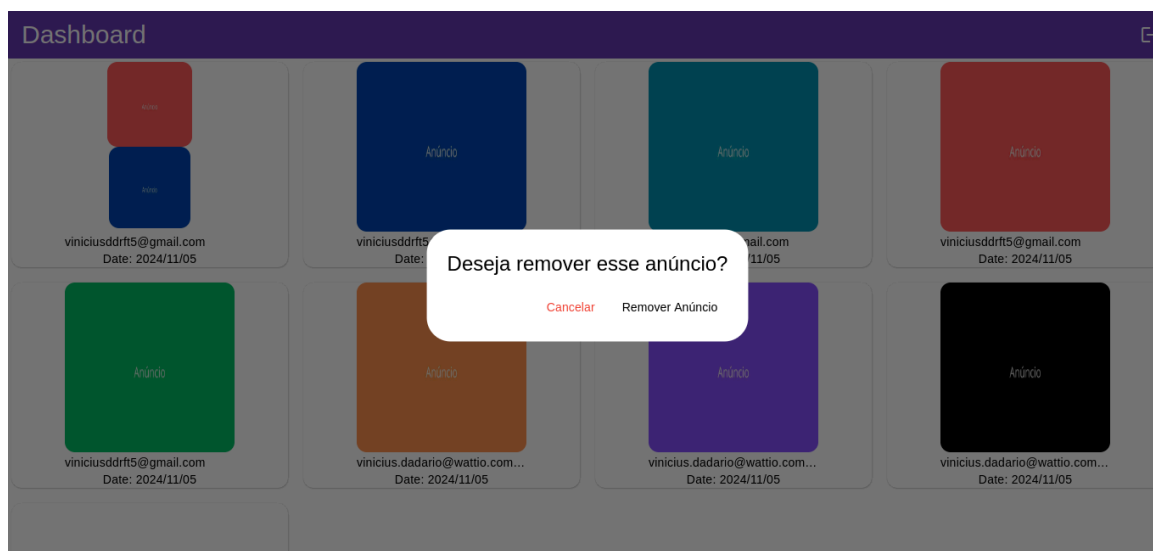
Figura 8 - Dashboard web



Fonte: Próprio Autor, 2024.

Na Figura 9, Tela do dashboard onde é mostrado o que acontece ao gesto de clicar em um anúncio, mostrando a opção de deletar o anúncio caso tenha as permissões necessárias.

Figura 9 - Demonstração de deletar anuncio web



Fonte: Próprio Autor, 2024.

3.4 Onboarding

Na Figura 10, Feature exclusiva do mobile, nessa tela é possível adicionar um anúncio selecionando uma imagem e apertando o botão de adicionar Anúncio, ou alterar para adicionar um Anúncio de 2 imagens.

Figura 10 - Adicionar anúncio de 1 imagem (usuário comum)



Fonte: Próprio Autor, 2024.

Na Figura 11, um recurso exclusivo da versão mobile, nessa tela é possível adicionar um anúncio de 2 imagens, onde é possível selecionar duas imagens e fazer upload do anúncio, isso na experiência de um usuário comum.

Figura 11 - Adicionar anúncio de 2 imagens (usuário comum)



A Figura 12 ilustra uma tela onde é possível adicionar um anúncio de 1 imagem, isso na experiência de um usuário administrador, podendo escolher o tempo de cada anúncio.

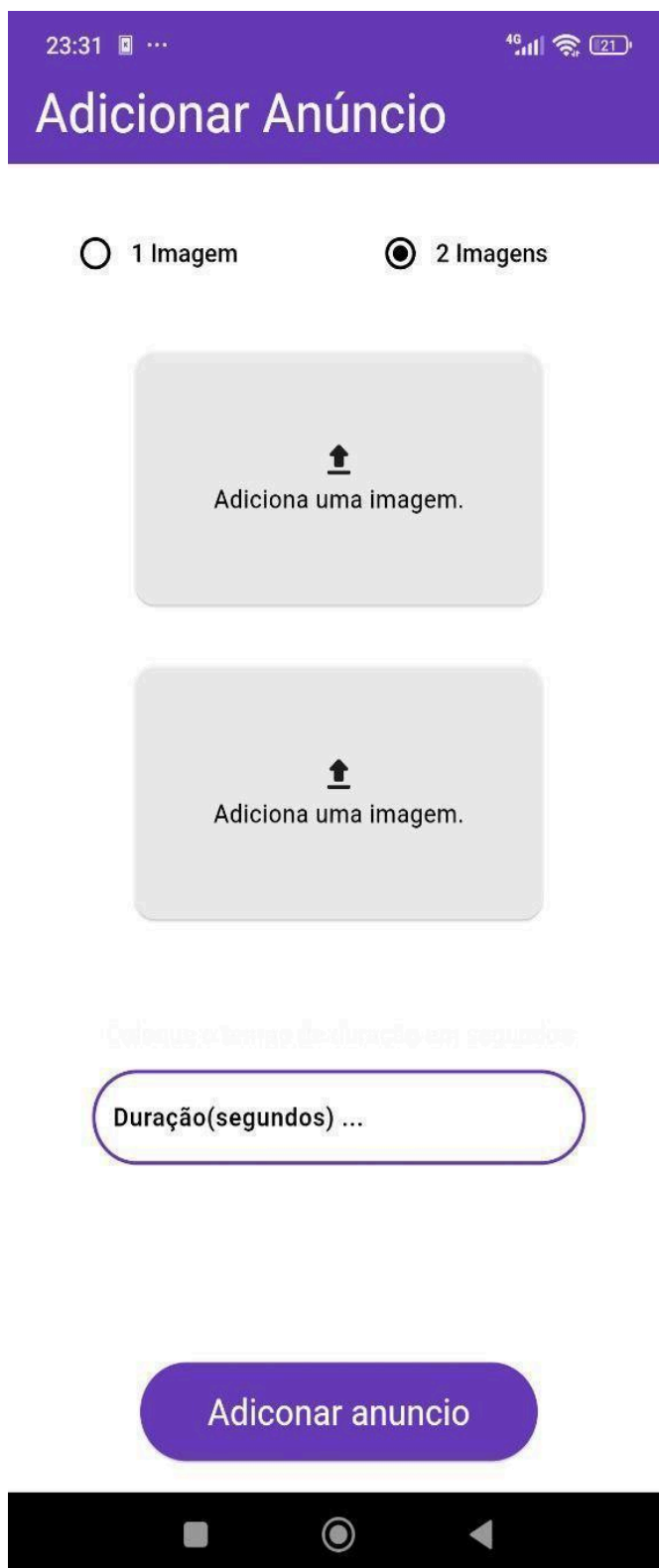
Figura 12 - Adicionar anúncio de 1 imagem (usuário administrador)



Fonte: Próprio Autor, 2024.

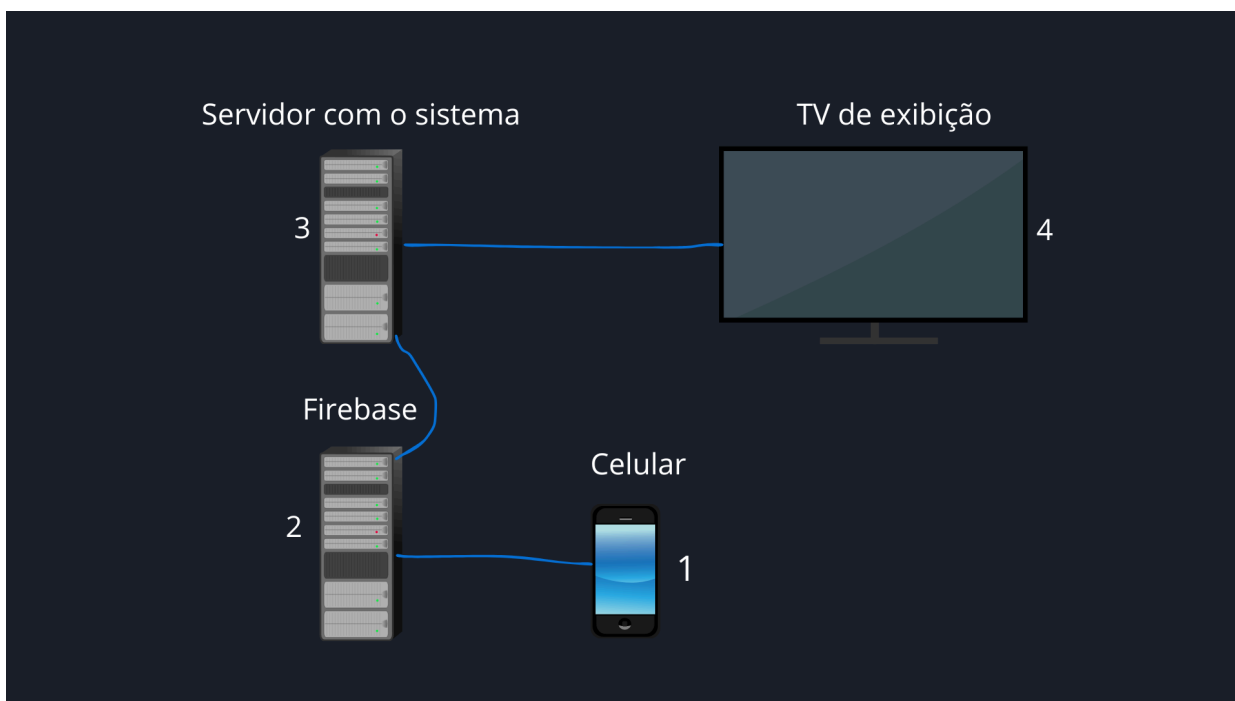
Na Figura 13, nessa tela é possível adicionar um anúncio de 2 imagens, onde é possível selecionar duas imagens e fazer upload do anúncio, isso na experiência de um usuário administrador, podendo escolher o tempo de cada anúncio.

Figura 13 - Adicionar anúncio de 2 imagens (usuário administrador)



3.5 Fluxo de dados final

Figura 14, aqui é mostrado o fluxo de dados final, mantendo as regras de segurança.



3.6 Home

Figura 15, recurso exclusivo da web onde os anúncios são exibidos e atualizados a cada 5 minutos via Firebase, removendo anúncios excluídos e adicionando novos rapidamente.

Figura 14 - Demonstração dos anúncios

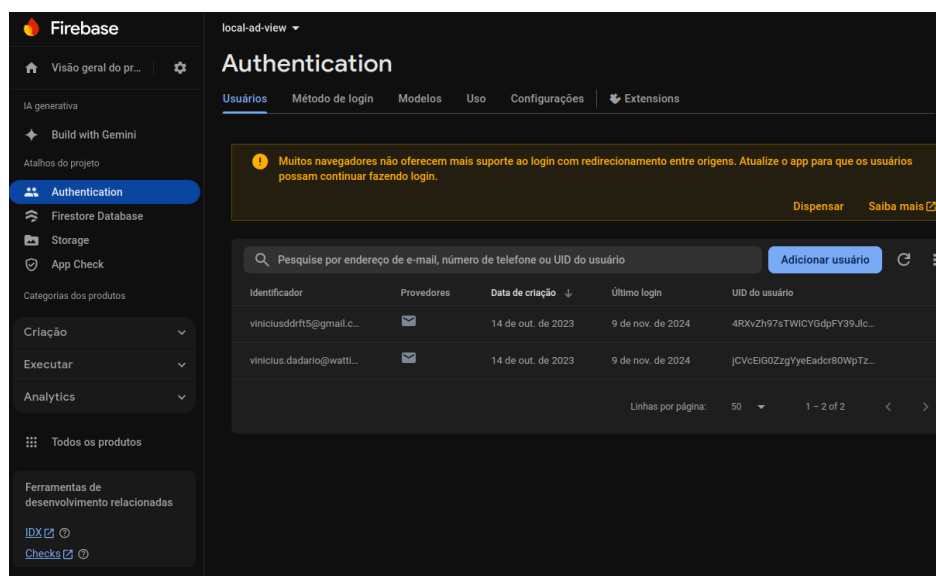


Fonte: Próprio Autor, 2024.

3.7 Firebase Authentication

Figura 16, uma das funcionalidades do Firebase utilizadas foi a autenticação. O Firebase simplifica o processo de login entregando todo backend para isso, proporcionando uma solução eficiente e fácil. Neste projeto, o método de autenticação escolhido foi o login com e-mail e senha.

Figura 15 - Demonstração do Firebase Authentication

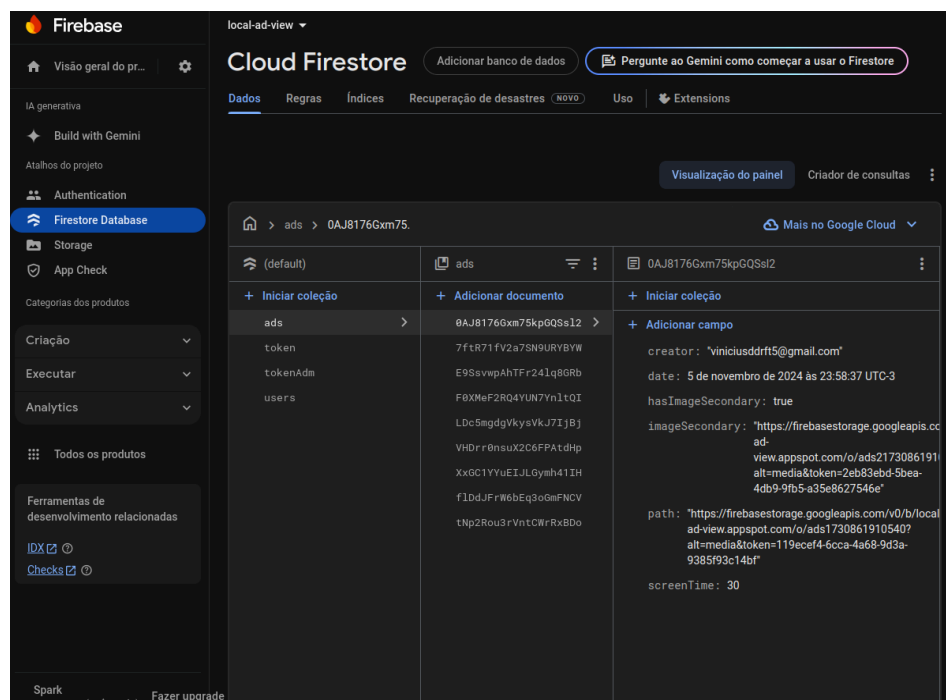


Fonte: Próprio Autor, 2024.

3.8 Firestore Database

Figura 17, o Firestore foi utilizado no aplicativo para armazenar informações relacionadas aos anúncios, dados dos usuários cadastrados e os tokens que diferenciam usuários comuns de administradores.

Figura 16 - Demonstração do Firestore



Fonte: Próprio Autor, 2024.

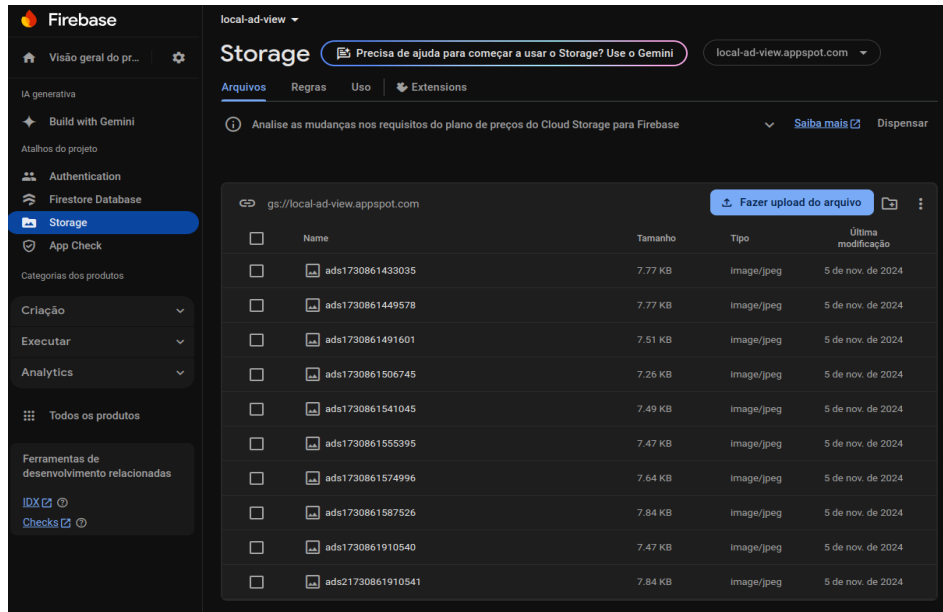
3.9 Firebase Storage

Figura 18, o Firebase Storage é utilizado para armazenar as imagens, enquanto o link gerado para cada imagem é salvo no Firestore, facilitando sua referência e acesso no aplicativo.

3.10 IDE

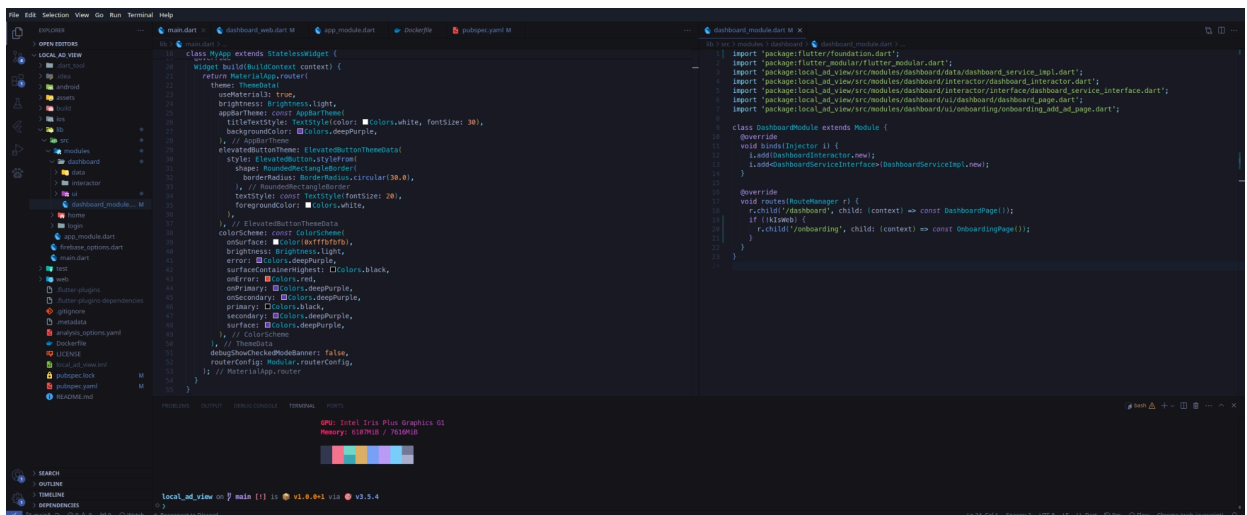
Figura 19, A IDE usada para o desenvolvimento foi o vscode, escolhida por ser leve e uma das IDEs que o flutter/dart tem plugins que dão suporte ao desenvolvimento de forma simples.

Figura 17 - Demonstração do Storage



Fonte: Próprio Autor, 2024.

Figura 18 - Ambiente de Desenvolvimento



Fonte: Próprio Autor, 2024.

4 CONCLUSÃO

O desenvolvimento deste aplicativo atingiu seu objetivo principal: oferecer uma maneira simples de subir anúncios em uma plataforma mobile, facilitando todo o processo. Ele respeita as regras de acesso da versão web, que opera em uma intranet e não pode ser acessada externamente. A aplicação conseguiu proporcionar a facilidade de uso e as medidas

de segurança esperadas. Além disso, este trabalho contribui significativamente para a otimização do gerenciamento de anúncios, permitindo que empresas e administradores centralizem e organizem suas campanhas de forma prática e eficiente. A possibilidade de gerenciar anúncios remotamente em dispositivos móveis reduz custos operacionais e aumenta a agilidade no processo, tornando-se uma ferramenta indispensável para ambientes corporativos modernos.

REFERÊNCIAS

DART. **Conheça o Dart**. Disponível em: <https://dart.dev>. Acesso em: 9 nov. 2024.

BAUER, Christine. **Interactive Digital Signage - An Innovative Service and Its Future Strategies: saiba mais sobre o Digital Signage**. 2011. Disponível em: <https://ieeexplore.ieee.org/abstract/document/6076433>. Acesso em: 18 nov. 2024.

FIREBASE. **Conheça o Firebase**. Disponível em: <https://firebase.google.com>. Acesso em: 9 nov. 2024.

FLUTTER. **Conheça o Flutter**. Disponível em: <https://flutter.dev>. Acesso em: 9 nov. 2024.

PODMAN. **Conheça o Podman**. Disponível em: <https://podman.io>. Acesso em: 10 nov. 2024.

GUERON, Shay. **SHA-512/256: saiba mais sobre o SHA-512**. 2011. Disponível em: <https://ieeexplore.ieee.org/abstract/document/5945260/authors#authors>. Acesso em: 18 nov. 2024.

VISUAL STUDIO CODE. **Conheça o Visual Studio Code**. Disponível em: <https://code.visualstudio.com>. Acesso em: 18 nov. 2024.