

UNIVERSIDADE RUY BARBOSA

DEIVIDE MACIEL - 202308426028

SLINK

Engenharia de Software

SALVADOR - BAHIA
2024

DEIVIDE MACIEL - 202308426028

SLINK

Engenharia de Software

Trabalho apresentado à disciplina Engenharia de
Software, da Universidade Ruy Barbosa.

Orientador(a): Prof. Heleno Filho.

SALVADOR - BAHIA

2024

SUMÁRIO

1. INTRODUÇÃO

Este projeto tem como objetivo apresentar as etapas do ciclo de desenvolvimento de uma aplicação web seguindo as diretrizes da Engenharia de Software. O sistema foi criado para solucionar a necessidade de gerenciamento de links. A metodologia adotada incluiu o uso de diversas técnicas recomendadas pela Engenharia de Software, englobando planejamento, avaliação e análise de viabilidade em colaboração com o cliente para garantir que suas expectativas fossem atendidas.

2. OBJETIVOS DA APLICAÇÃO

Este projeto tem como objetivo desenvolver um sistema de gerenciamento de links. O sistema foi concebido para atender a uma demanda de organizar e gerenciar links de forma eficiente. Com funcionalidades como login, adição e exclusão de links, criação e gerenciamento de pastas, o sistema oferece uma solução robusta e intuitiva para o gerenciamento de links.

3. ELICITAÇÃO DOS REQUISITOS:

Para a fase de levantamento de requisitos foram realizadas entrevistas e observações do cenário real para obter uma compreensão mais precisa das

necessidades do cliente e dos processos envolvidos no projeto. O objetivo era obter uma visão geral do projeto e sua aplicação no ambiente real, identificando todos os processos relevantes.

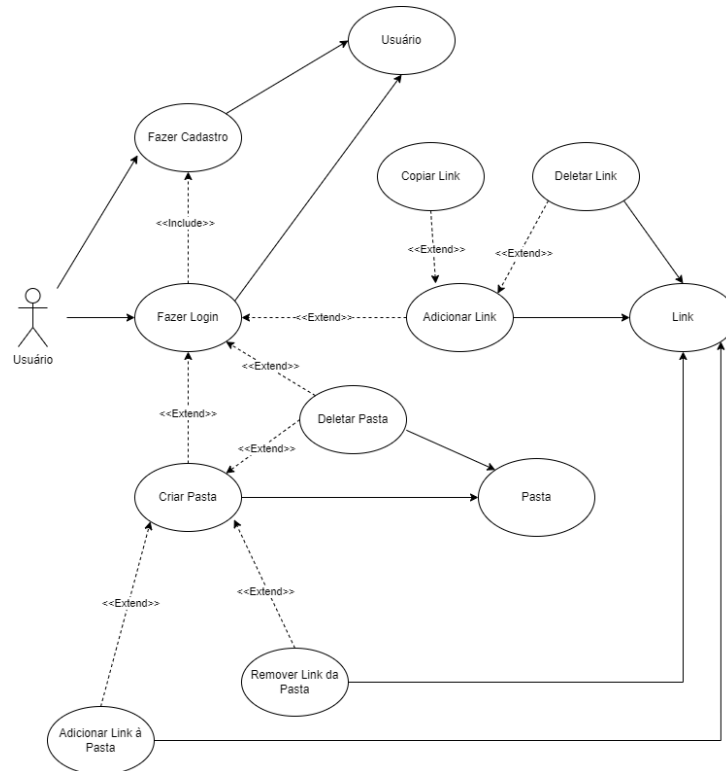
3.1. Levantamento de Requisitos

- **Requisitos Funcionais (RF):**

- **RF1: O usuário deve poder se cadastrar:** Os usuários devem poder se cadastrar na plataforma fornecendo um nome de usuário, e-mail e senha.
- **RF2: O usuário deve poder fazer login:** Os usuários devem poder fazer login na plataforma usando e-mail e senha.
- **RF3: O usuário deve poder adicionar um link:** Os usuários devem poder adicionar novos links à plataforma, fornecendo uma URL e uma descrição.
- **RF4: O usuário deve poder excluir um link:** Os usuários devem poder excluir links previamente adicionados à plataforma.
- **RF5: O usuário deve poder criar pastas:** Os usuários devem poder criar pastas para organizar seus links.
- **RF6: O usuário deve poder adicionar links a pastas:** Os usuários devem poder adicionar links a pastas criadas por eles.
- **RF7: O usuário deve poder remover links de pastas:** Os usuários devem poder remover links de pastas existentes.
- **RF8: O usuário deve poder excluir pastas:** Os usuários devem poder excluir pastas criadas por eles, juntamente com todos os links contidos nelas.

3. DIAGRAMA DE CASOS DE USO

O diagrama a seguir apresenta os diferentes casos de uso do sistema, delineando as ações que os usuários podem realizar, além de demonstrar a interação entre esses casos de uso e as entidades relacionadas do banco de dados.



- **Fazer Cadastro:** Permite que novos usuários criem uma conta na aplicação fornecendo informações como nome, e-mail e senha.
-
- **Fazer Login:** Permite que usuários existentes façam login na aplicação usando suas credenciais.
- **Adicionar Link:** Permite que os usuários adicionem novos links, fornecendo uma URL e uma descrição opcional.
- **Copiar Link:** Permite que os usuários copiem o link para a área de transferência.
- **Deletar Link:** Permite que os usuários removam links que não são mais necessários.
- **Criar Pasta:** Permite que os usuários criem novas pastas para organizar seus links.
- **Adicionar Link à Pasta:** Permite que os usuários adicionem links a pastas específicas para melhor organização.
- **Remover Link da Pasta:** Permite que os usuários removam links de pastas, sem deletar o link do sistema.

4. DIAGRAMA DE ENTIDADE-RELACIONAMENTO

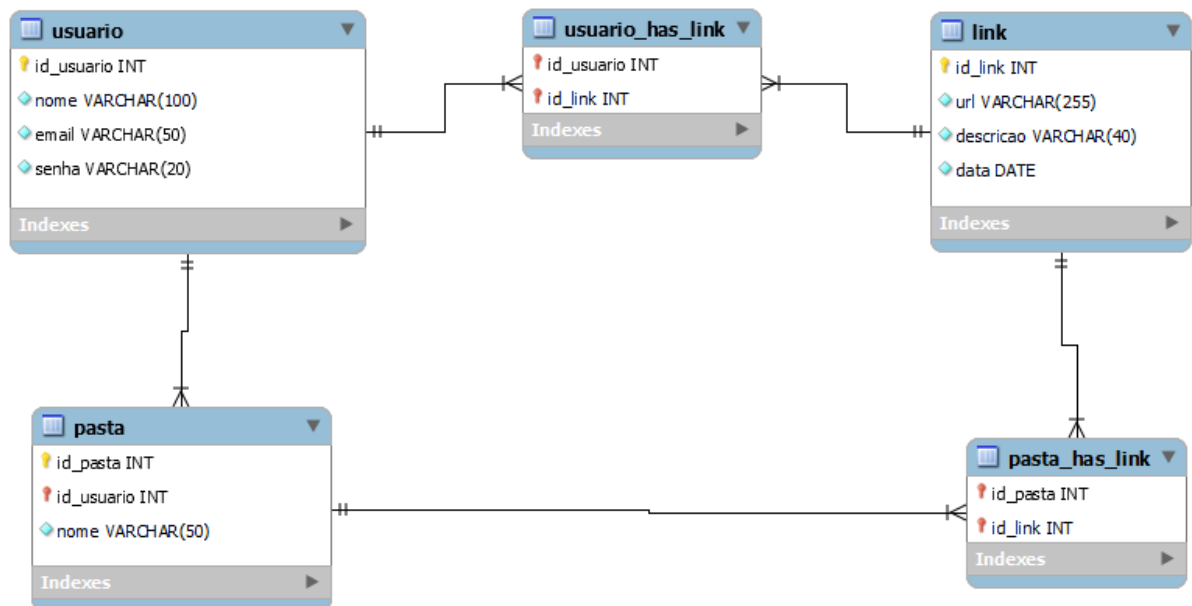
O diagrama a seguir apresenta a representação visual das entidades principais do sistema e as relações entre elas. Além de mostrar também o tipo de dados das entidades, chaves primárias e chaves estrangeiras.

4.1 Entidades envolvidas:

Usuário: Representa os usuários da aplicação. Cada usuário possui as informações nome, e-mail, e senha

Link: Representa os links salvos pelos usuários. Cada link inclui uma URL, uma descrição e a data que foi adicionado.

Pasta: Representa pastas criadas pelos usuários para organizar os links. Cada pasta tem um nome único por usuário e pode conter múltiplos links.



4.2 Dicionário de Dados (DDL)

```

1  -- MySQL Workbench Forward Engineering
2
3  SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4  SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5  SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
6
7
8  CREATE SCHEMA IF NOT EXISTS `Slink` DEFAULT CHARACTER SET utf8 ;
9  USE `Slink` ;
10
11  CREATE TABLE IF NOT EXISTS `Slink`.`usuario` (
12    `id_usuario` INT NOT NULL AUTO_INCREMENT,
13    `nome` VARCHAR(100) NOT NULL,
14    `email` VARCHAR(50) NOT NULL,
15    `senha` VARCHAR(20) NOT NULL,
16    PRIMARY KEY (`id_usuario`),
17    UNIQUE INDEX `email_UNIQUE` (`email` ASC) VISIBLE)
18  ENGINE = InnoDB;
19
20
21  CREATE TABLE IF NOT EXISTS `Slink`.`link` (
22    `id_link` INT NOT NULL AUTO_INCREMENT,
23    `url` VARCHAR(255) NOT NULL,
24    `descricao` VARCHAR(40) NOT NULL,
25    `data` DATE NOT NULL,
26    PRIMARY KEY (`id_link`))
27  ENGINE = InnoDB;
28
29  CREATE TABLE IF NOT EXISTS `Slink`.`usuario_has_link` (
30    `id_usuario` INT NOT NULL,
31    `id_link` INT NOT NULL,
32    PRIMARY KEY (`id_usuario`, `id_link`),
33    INDEX `fk_usuario_has_link_link1_idx` (`id_link` ASC) VISIBLE,
34    INDEX `fk_usuario_has_link_usuario_idx` (`id_usuario` ASC) VISIBLE,
35    CONSTRAINT `fk_usuario_has_link_usuario`
36      FOREIGN KEY (`id_usuario`)
37        REFERENCES `Slink`.`usuario` (`id_usuario`)
38        ON DELETE NO ACTION
39        ON UPDATE NO ACTION,
40    CONSTRAINT `fk_usuario_has_link_link1`
41      FOREIGN KEY (`id_link`)
42        REFERENCES `Slink`.`link` (`id_link`)
43        ON DELETE NO ACTION
44        ON UPDATE NO ACTION)
45  ENGINE = InnoDB;
46
47
48  CREATE TABLE IF NOT EXISTS `Slink`.`pasta` (
49    `id_pasta` INT NOT NULL AUTO_INCREMENT,
50    `id_usuario` INT NOT NULL,
51    `nome` VARCHAR(50) NOT NULL,
52    PRIMARY KEY (`id_pasta`, `id_usuario`),
53    INDEX `fk_pasta_usuario1_idx` (`id_usuario` ASC) VISIBLE,
54    UNIQUE INDEX `nome_UNIQUE` (`nome` ASC) VISIBLE,
55    CONSTRAINT `fk_pasta_usuario1`
56      FOREIGN KEY (`id_usuario`)
57        REFERENCES `Slink`.`usuario` (`id_usuario`)
58        ON DELETE NO ACTION
59        ON UPDATE NO ACTION)
60  ENGINE = InnoDB;
61
62
63  -- -----
64  -- Table `Slink`.`pasta_has_link`
65  -- -----
66  CREATE TABLE IF NOT EXISTS `Slink`.`pasta_has_link` (
67    `id_pasta` INT NOT NULL,
68    `id_link` INT NOT NULL,
69    PRIMARY KEY (`id_pasta`, `id_link`),
70    INDEX `fk_pasta_has_link_link1_idx` (`id_link` ASC) VISIBLE,
71    INDEX `fk_pasta_has_link_pasta1_idx` (`id_pasta` ASC) VISIBLE,
72    CONSTRAINT `fk_pasta_has_link_pasta1`
73      FOREIGN KEY (`id_pasta`)
74        REFERENCES `Slink`.`pasta` (`id_pasta`)
75        ON DELETE NO ACTION
76        ON UPDATE NO ACTION,
77    CONSTRAINT `fk_pasta_has_link_link1`
78      FOREIGN KEY (`id_link`)
79        REFERENCES `Slink`.`link` (`id_link`)
80        ON DELETE NO ACTION
81        ON UPDATE NO ACTION)
82  ENGINE = InnoDB;
83
84
85  SET SQL_MODE=@OLD_SQL_MODE;
86  SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
87  SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
88

```

5. Política de Implantação

A política de implantação do projeto de gerenciamento de links visa garantir que o processo de entrega e manutenção da aplicação seja eficiente, seguro e confiável. Utilizando as ferramentas Vercel para hospedagem e Git/GitHub para controle de versão e CI/CD (Continuous Integration/Continuous Deployment), a implantação será realizada de forma automatizada, assegurando qualidade e consistência em cada atualização.

Ferramentas Utilizadas

- **Vercel:** Plataforma de hospedagem que oferece implantação contínua e integração com GitHub, facilitando a distribuição da aplicação.
- **Git:** Sistema de controle de versão distribuído para gerenciamento de código-fonte.
- **GitHub:** Repositório Git baseado na web que fornece controle de versão e funcionalidades de CI/CD.

Fluxo de Trabalho de Desenvolvimento

1. **Controle de Versão com Git e GitHub:**
 - Todo o código-fonte do projeto será armazenado e versionado no GitHub.
 - O desenvolvimento será organizado em branches, seguindo uma estratégia de branching, como GitFlow.
 - Branches principais:
 - **main:** Contém o código estável e pronto para produção.
 - **develop:** Contém o código em desenvolvimento e integração contínua.
 - Branches de funcionalidade, correção e hotfix serão criadas a partir de **develop** e mescladas de volta após revisão e testes.

Integração Contínua (CI):

- A cada commit realizado em qualquer branch, o GitHub Actions será acionado para executar testes automatizados e validação do código.
- Scripts de build e testes serão configurados para garantir que o código seja construído e testado corretamente antes de qualquer integração.

Deploy Contínuo (CD)

1. **Implantação em Vercel:**
 - Vercel será configurado para realizar implantações automáticas a partir das branches **main** e **develop**

- Deploy de `develop` será direcionado para o ambiente de staging, permitindo a validação das mudanças antes da produção.
- Deploy de `main` será direcionado para o ambiente de produção.

Revisão de Código:

- Todo código a ser integrado na branch `develop` ou `main` deve passar por um processo de revisão de código via Pull Requests (PRs).
- Revisores devem verificar a qualidade do código, a conformidade com os padrões de codificação e a cobertura de testes.

6. Protótipo da Aplicação



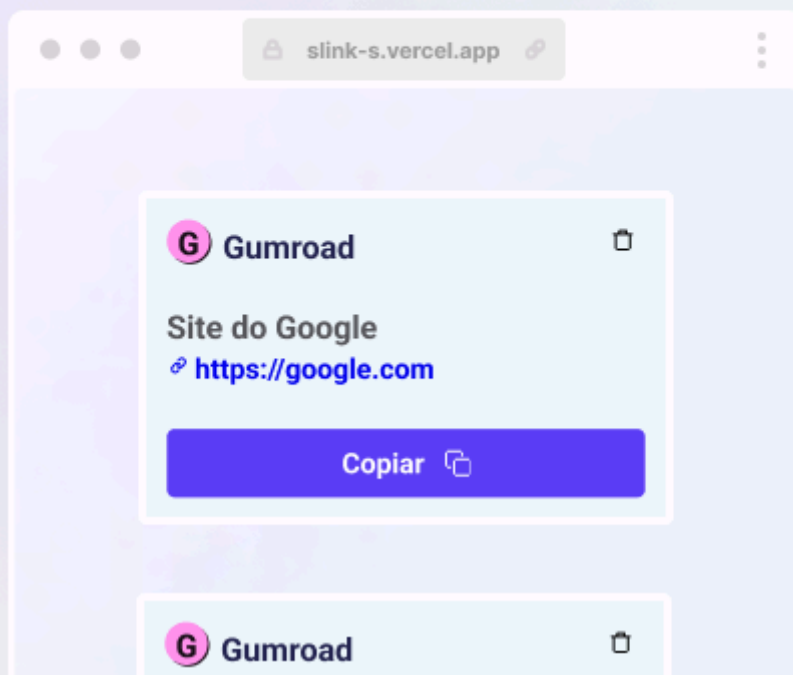
Entrar

Welcome to Slink

Complete link management

Your ultimate link management platform to save, organize, and access your internet links with ease.

Criar conta





Para continuar, faça login no Slink.

E-mail

Senha

Esqueceu a sua senha?

Entrar

Não tem uma conta?

CRIAR UM ACONTA NO SLINK



**Para continuar, crie uma conta no
Slink**

Nome

E-mail

E-mail

E-mail

Senha

Senha

Entrar

Já tem uma conta?

ENTRAR NO SLINK



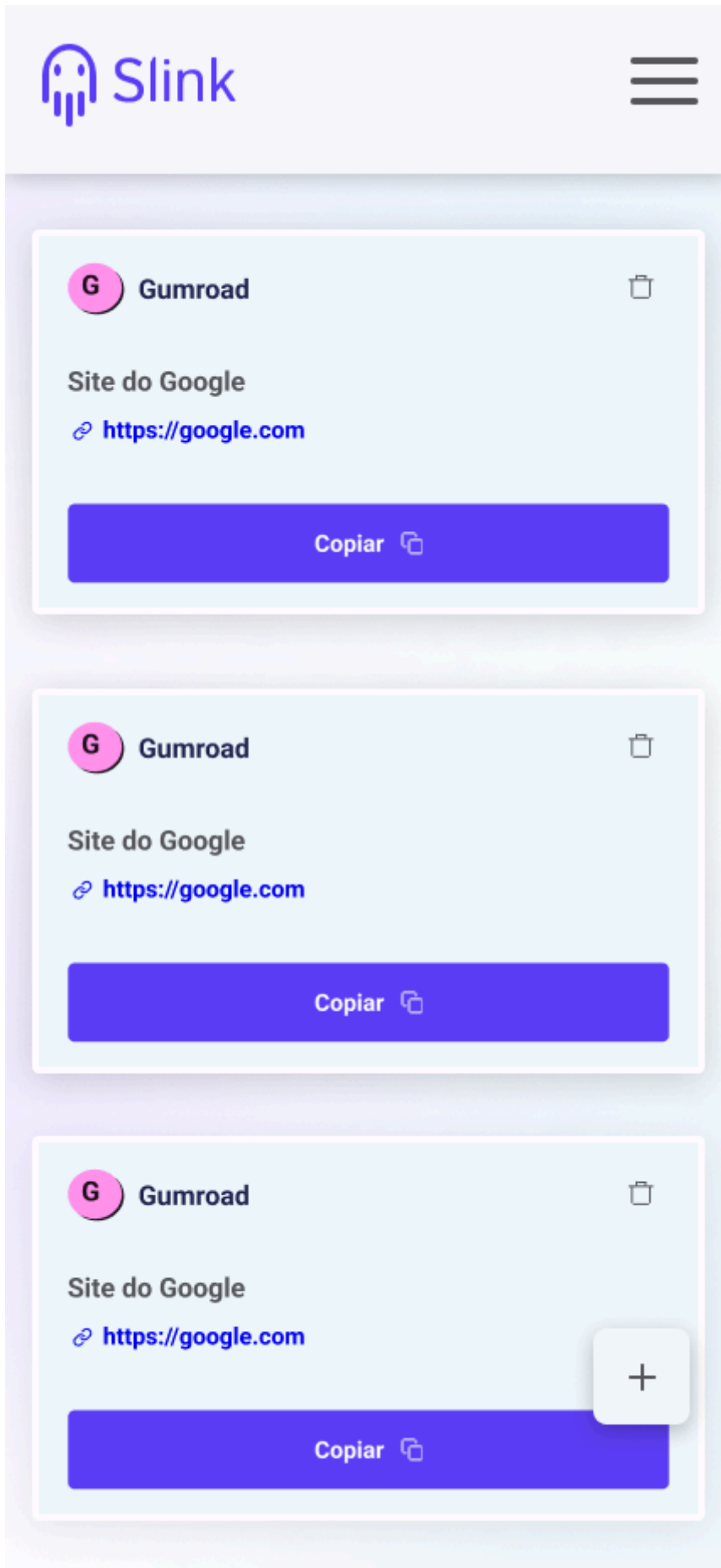
Recuperar senha


Código

Nova senha

Confirmar nova senha


Continuar



 **Slink**

G Gumroad

Site do Google

 **Título**

Link


Descrição

Adicionar


Cancelar


G Gumroad

Site do Google

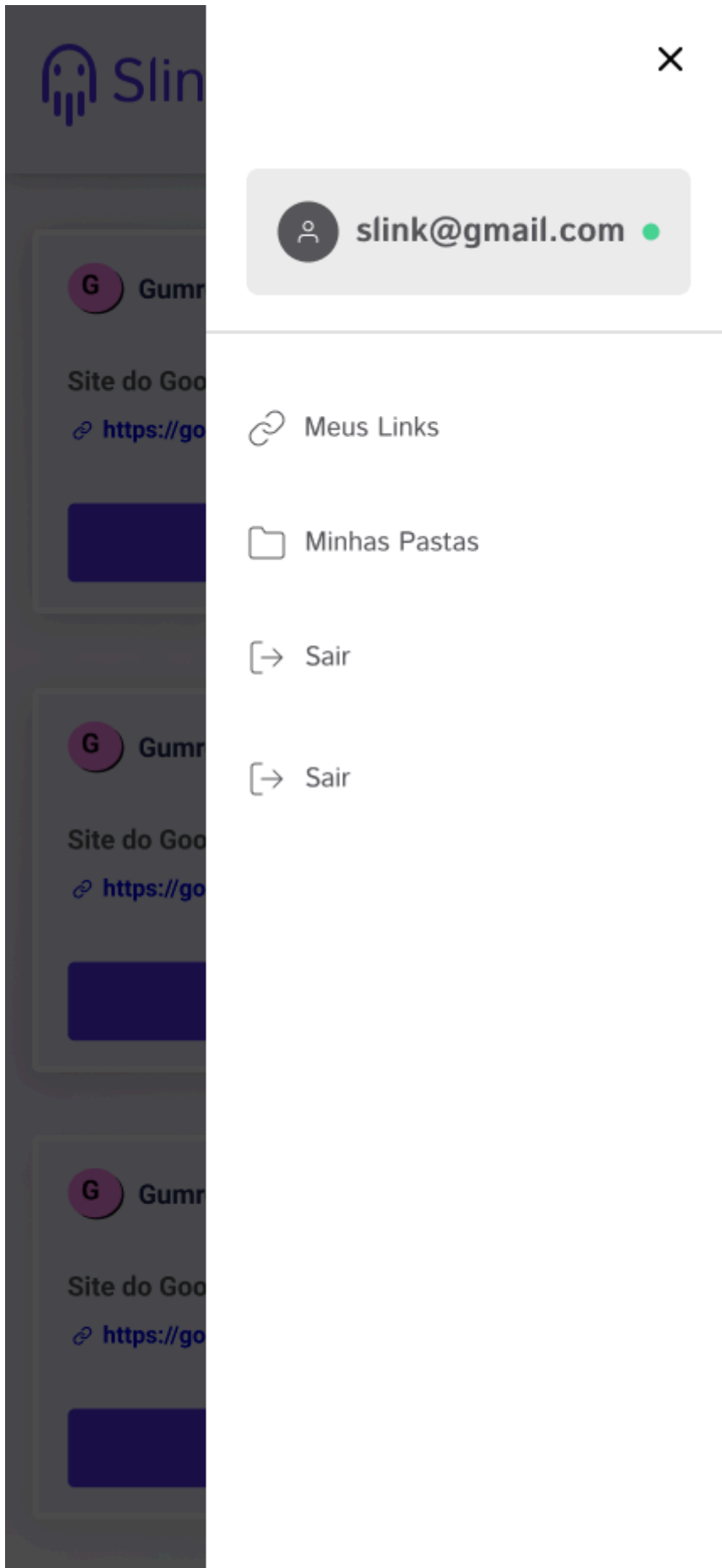
 <https://google.com>

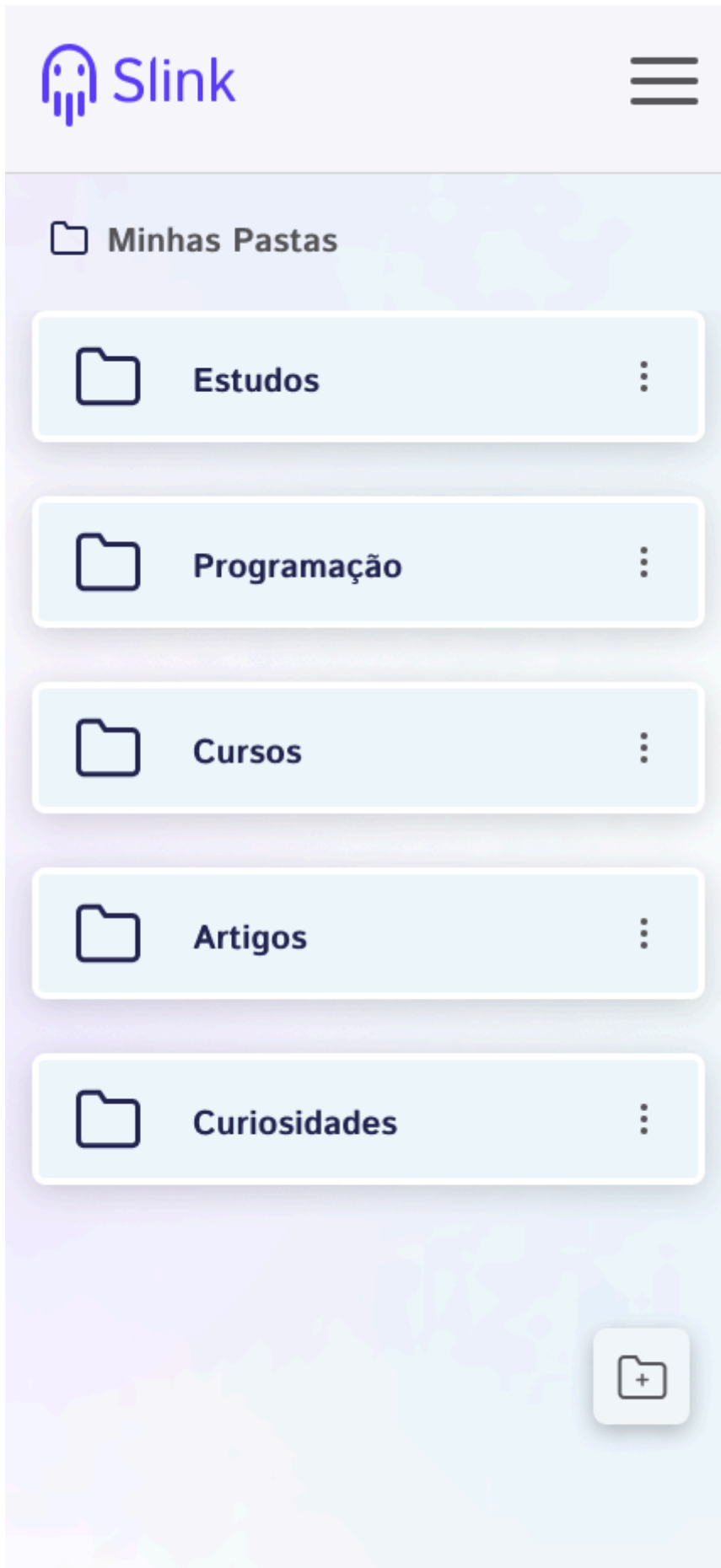
Copiar

















 **Slink**



 **Minhas Pastas**

 **Estudos** 

 **Título**

Nome

Descrição

Criar **Cancelar**

