



ESCOLA SENAI “José Polizotto”
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS

BEATRIZ BEZERRA BARBOSA

ZEN HABIT

GARÇA

2025

ESCOLA SENAI “José Polizotto”
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS

BEATRIZ BEZERRA BARBOSA

ZEN HABIT

Projeto apresentado à Escola SENAI “José Polizotto”
de Garça como parte dos requisitos para obtenção do
título de Técnico em Desenvolvimento de Sistemas

Orientador: Prof. Ricardo Fonseca da Cruz.

Orientador: Prof. Robson Ramos da Silva.

GARÇA

2025

Barbosa, Beatriz Bezerra

Zen Habit / Beatriz Bezerra Barbosa; ...[et al.]. Garça, 2025. 32 p.

Orientador: Prof. Ricardo Fonseca da Cruz

Orientador: Prof. Robson Ramos da Silva

Trabalho de Conclusão de Curso (Técnico) - Serviço Nacional de Aprendizagem Industrial,
Curso Técnico de Desenvolvimento de Sistemas, 2025

Zen Habit; aplicativo; gerenciamento de hábitos;

Barbosa, Beatriz.

BEATRIZ BEZERRA BARBOSA

ZEN HABIT

Trabalho de Conclusão de Curso apresentado à Escola SENAI “José Polizotto” de Garça como parte dos requisitos para obtenção do título de Técnico em Desenvolvimento de Sistemas.

Garça, 5 de Dezembro de 2025

Banca Examinadora:

Prof.Ricardo Fonseca da Cruz

Prof.Robson Ramos da Silva

DEDICATÓRIA

Dedico este trabalho à minha família, pelo incentivo, apoio incessante e presença ao longo de toda a minha trajetória acadêmica, os quais foram essenciais para a concretização deste projeto.

Dedico também a mim mesma, pela dedicação, constância diante dos desafios e determinação em alcançar todos os meus objetivos.

AGRADECIMENTOS

Meus agradecimentos a todos os que colaboraram, direta ou indiretamente, para a concretização deste trabalho, especialmente aos meus professores Ricardo Fonseca e Robson Ramos, pela orientação, dedicação e apoio ao longo do curso e em todas as etapas deste projeto, cujos conhecimentos transmitidos foram indispensáveis para sua realização.

Aos meus amigos e colegas, pela cumplicidade, apoio e troca de saberes durante o desenvolvimento do projeto.

“Qualquer tecnologia suficientemente avançada é
indistinguível da magia.”

Arthur C. Clarke

RESUMO

O projeto ZenHabit visa auxiliar os usuários na criação, monitoramento e manutenção de hábitos voltados à saúde e ao bem-estar por meio de uma aplicação mobile. O aplicativo oferece funcionalidades de criação, edição, exclusão e visualização de hábitos, tornando o acompanhamento contínuo mais organizado e motivador. A metodologia utilizada envolveu a análise das necessidades dos usuários e padrões observados, permitindo a implementação de recursos como ícones relacionados aos hábitos, definição de frequência e monitoramento do cumprimento das atividades. Dessa maneira, o aplicativo facilita o acompanhamento para o desenvolvimento pessoal, mesmo diante da correria cotidiana. Conclui-se que a tecnologia pode ser um mecanismo eficiente para impulsionar mudanças positivas no estilo de vida dos usuários.

Palavras-chave: Hábitos saudáveis, Aplicativo mobile, Monitoramento, Desenvolvimento pessoal.

ABSTRACT

The ZenHabit project aims to assist users in creating, monitoring, and maintaining habits focused on health and well-being through a mobile application. The app offers functionalities for creating, editing, deleting, and viewing habits, making continuous monitoring more organized and motivating. The methodology used involved analyzing user needs and observed patterns, allowing the implementation of features such as icons related to habits, frequency definition, and monitoring of activity completion. In this way, the application facilitates monitoring for personal development, even amidst the daily rush. It is concluded that technology can be an efficient mechanism to drive positive changes in users' lifestyles.

Keywords: Healthy habits, Mobile application, Monitoring, Personal development.

LISTA DE FIGURAS

FIGURA 1 - IDE VSCODE	16
FIGURA 2 - API ROTAS COM MÉTODOS HTTP	18
FIGURA 3 - BANCO DE DADOS	19
FIGURA 4 - TRELLO.....	20
FIGURA 5 - FIGMA	21
FIGURA 6 – GIT E GITHUB	22
FIGURA 7 - DIAGRAMA DO BANCO DE DADOS	30
FIGURA 8 – PROTOTIPAÇÃO DO FIGMA	31
FIGURA 9 - PROTOTIPAÇÃO DO FIGMA	32

SUMÁRIO

1.INTRODUÇÃO	12
2.DESENVOLVIMENTO	13
2.1 Objetivo Geral	13
2.2 Objetivos específicos	13
3.REVISÃO BIBLIOGRÁFICA.....	14
4.METODOLOGIA	15
5.MÉTODOS	16
5.1 VsCode	16
5.2 Framework.....	16
5.2.1 Node.js	17
5.2.2 React Native.....	17
5.3 API.....	18
5.4 Banco de Dados	19
5.6 JavaScript	19
5.7 Trello	20
5.8 Figma.....	20
5.9 Git e GitHub	21
5.10 Insomnia.....	22
6 RESULTADOS.....	23
7 CONCLUSÃO	24
8.REFERÊNCIAS	25
APÊNDICE A – BIBLIOTECAS UTILIZADAS NO PROJETO FRONT-END	27
APÊNDICE B – BIBLIOTECAS UTILIZADAS NO PROJETO BACK-END.....	29
APÊNDICE C – DIAGRAMA DO BANCO DE DADOS	30
APÊNDICE D – PROTOTIPAÇÃO DAS TELAS DO APP.....	31

1.INTRODUÇÃO

Hodiernamente, tem-se notado o crescente interesse por um estilo de vida mais saudável. A busca pela implementação de hábitos saudáveis cotidianamente tem transformado de maneira significativa a forma como lidamos com o próprio autocuidado. Segundo o Hospital Israelita Albert Einstein (2025), “um dos grandes benefícios de investir em hábitos mais saudáveis está na melhoria do estado geral da saúde. Quando temos uma rotina mais equilibrada e repleta de tarefas benéficas, acabamos ganhando em melhorias para o nosso organismo”.

Nesse contexto, observa-se que a tecnologia tem desempenhado um papel fundamental no incentivo à adoção e manutenção desses hábitos, tornando-se uma aliada na promoção da saúde e do bem-estar. Aplicativos de monitoramento pessoal surgem como ferramentas eficazes para auxiliar os indivíduos a acompanharem sua rotina e alcançarem metas de forma mais organizada e motivadora.

Segundo a OECD (2024), as tecnologias digitais têm potencial para influenciar vários domínios da vida humana, incluindo trabalho, consumo e interação social, ainda que seus impactos no bem-estar global permaneçam complexos. A transição para plataformas digitais está reformulando o cenário da organização pessoal, com soluções digitais proporcionando uma forma mais eficiente de planejar, organizar e cumprir tarefas e hábitos individuais.

Dentro desse âmbito, o aplicativo de gestão de hábitos digitais ZenHabit será disponibilizado para sistemas operacionais Android e iOS. Sua missão é facilitar o acompanhamento e o controle de hábitos voltados à saúde e ao bem-estar, oferecendo uma interface intuitiva e recursos que incentivam a constância e o desenvolvimento pessoal. A tecnologia aplicada no projeto tem o potencial de atender às necessidades dos usuários de maneira moderna, eficiente e acessível, promovendo uma rotina mais equilibrada e produtiva, em sintonia com as tendências emergentes no campo digital.

2.DESENVOLVIMENTO

O aplicativo ZenHabit foi desenvolvido com o intuito de auxiliar os usuários na organização de hábitos pessoais por meio de uma interface de fácil navegação. A aplicação visa incentivar práticas positivas e constantes, promovendo o equilíbrio entre produtividade, autocuidado e qualidade de vida.

A proposta do projeto é oferecer uma ferramenta intuitiva, moderna e funcional que permita cadastrar, editar, excluir e visualizar hábitos de forma simples e organizada, além de um sistema de histórico que possibilita ao usuário acompanhar seu progresso e manter a motivação ao longo do tempo.

Ao proporcionar a organização de hábitos e a promoção de um estilo de vida saudável, o ZenHabit responde à crescente preocupação com a qualidade de vida. O desenvolvimento ágil com React Native possibilitou a criação de um aplicativo acessível e eficiente, que facilita o cumprimento de hábitos. Essa abordagem tecnológica garante que o aplicativo permaneça atualizado e adaptado às necessidades dos usuários.

Dessa forma, o ZenHabit demonstra como a tecnologia pode ser uma aliada na construção de uma rotina mais equilibrada e saudável. Combinando recursos visuais atrativos com funcionalidades práticas, o aplicativo promove a organização pessoal e o autoconhecimento, evidenciando o potencial das soluções digitais como instrumentos eficazes de apoio ao bem-estar e à qualidade de vida.

2.1 Objetivo Geral

Desenvolver um aplicativo mobile, chamado ZenHabit, com o propósito de auxiliar os usuários na criação, organização e acompanhamento de hábitos saudáveis, promovendo o equilíbrio entre produtividade, autocuidado e qualidade de vida.

2.2 Objetivos específicos

Desenvolver uma interface intuitiva que permita ao usuário criar, editar e acompanhar hábitos de forma prática, com um sistema de histórico que registre o progresso e incentive a constância, proporcionando um design moderno e funcional que motive o autocuidado e a organização pessoal.

3.REVISÃO BIBLIOGRÁFICA

O propósito da pesquisa para o desenvolvimento deste aplicativo foi obter informações detalhadas sobre as melhores práticas de construção de sistemas digitais de acompanhamento de hábitos, considerando a experiência do usuário e a eficácia no monitoramento das atividades diárias. Artigos científicos, livros e documentações relacionadas ao desenvolvimento de aplicativos móveis, design de interfaces, sistemas de gestão de hábitos e tecnologias afins foram analisados, a fim de garantir a implementação de soluções inovadoras, eficientes e alinhadas às necessidades dos usuários.

4.METODOLOGIA

Para o desenvolvimento dos métodos do projeto, baseou-se na realização de consultas em diversas fontes científicas e documentações técnicas, com o objetivo de compreender as principais tecnologias utilizadas, destacando suas aplicações, funcionalidades e usabilidade.

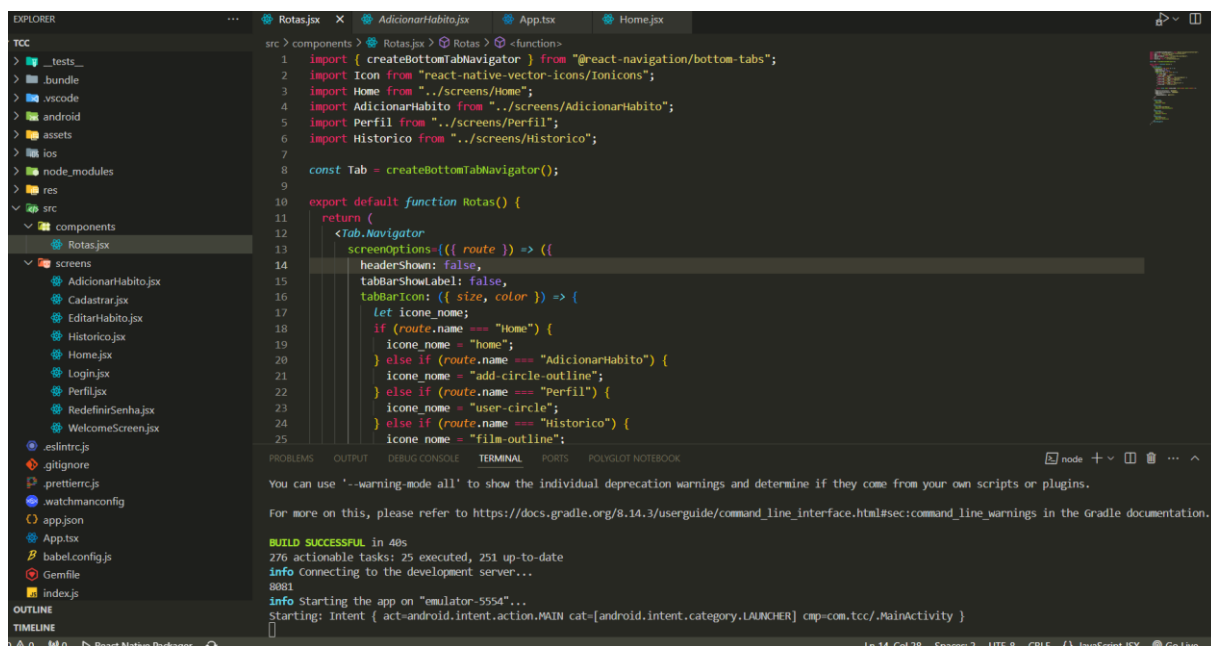
5.MÉTODOS

5.1 VsCode

De acordo com a documentação oficial da Microsoft (2023), o Visual Studio Code é um editor de código leve, porém robusto, compatível com diferentes sistemas operacionais, como Windows, macOS e Linux. A ferramenta oferece suporte para depuração, controle de versionamento com Git, realce de sintaxe e diversas extensões que permitem personalização, o que a torna amplamente utilizada por desenvolvedores em todo o mundo.

A Figura 1 ilustra um exemplo do Visual Studio Code utilizado durante o desenvolvimento do projeto. A escolha dessa ferramenta se deu por sua popularidade entre os desenvolvedores, além de sua estrutura flexível e desempenho confiável, que contribuíram significativamente para a execução do trabalho.

Figura 1 - IDE VsCode



Fonte: A autora

5.2 Framework

Conforme a documentação oficial da Django Software Foundation (2024), “um framework web é uma estrutura que fornece componentes prontos para lidar com tarefas comuns de desenvolvimento.” Dessa forma, um framework pode ser compreendido como um conjunto de bibliotecas e ferramentas voltadas ao desenvolvimento de aplicações, criadas para

solucionar problemas recorrentes e permitir a reutilização de código. A utilização de frameworks no desenvolvimento deste projeto mostrou-se essencial, pois oferece recursos pré-construídos que facilitam a implementação de boas práticas de codificação, promovendo organização, padronização e eficiência. Além disso, o uso dessa estrutura contribuiu para acelerar o processo de desenvolvimento, garantindo consistência e qualidade na construção das funcionalidades do aplicativo.

5.2.1 Node.js

O Node.js foi adotado como framework principal para a construção da API RESTful que compõe o back-end da aplicação. Conforme a documentação oficial do Node.js (2023), “Node.js é um ambiente de execução JavaScript construído no motor V8 do Chrome, ideal para criar aplicações de rede escaláveis e eficientes.” Por se tratar de uma tecnologia consolidada no mercado, de fácil aplicação e com tipagem dinâmica e ágil, o Node.js se destaca por seu código aberto, que possibilita a criação e disseminação de módulos desenvolvidos pela comunidade global.

A escolha dessa tecnologia deve-se ao seu alto desempenho, escalabilidade e repleta gama de bibliotecas e ferramentas que aprimoram o processo de desenvolvimento. Além disso, o Node.js permite a execução de código JavaScript no lado do servidor, tornando viável o desenvolvimento de aplicações autônomas e modernas.

5.2.2 React Native

De acordo com a documentação oficial do React Native, esse framework de código aberto, mantido pela empresa Meta, permite o desenvolvimento de aplicações móveis para Android e iOS a partir de uma única base de código escrita em JavaScript. Essa tecnologia foi projetada com a finalidade de unir a eficiência do desenvolvimento web à performance do desenvolvimento nativo, permitindo que componentes escritos em JavaScript se comuniquem diretamente com as APIs das plataformas móveis. Tornando possível desenvolver interfaces modernas e responsivas, sem abrir mão da experiência de uso característica dos aplicativos nativos.

Este framework é muito útil para empresas e desenvolvedores que visam otimizar tempo e reduzir custos de produção, assegurando a qualidade e a performance dos aplicativos nativos. Além disso, o React Native garante a consistência visual e funcional das aplicações,

em sintonia na possibilidade de reutilização de grande parte do código entre diferentes plataformas de desenvolvimento .

Sua habilidade de renderizar interfaces de forma nativa nos sistemas iOS e Android, juntamente à utilização de bibliotecas externas e plugins integrados, possibilita uma alternativa poderosa para o desenvolvimento multiplataforma.

5.3 API

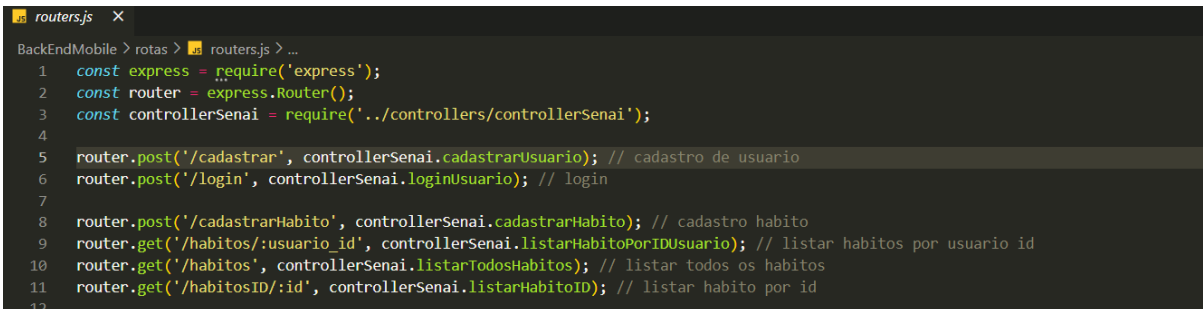
De acordo com Sousa (2000), “a API é composta por um conjunto de primitivas que integram a definição e a manipulação de objetos em uma representação compatível com uma linguagem de programação orientada a objetos.” Dessa maneira, a API (Application Programming Interface) possui como fundamento o estabelecimento e a comunicação entre o cliente e o servidor, possibilitando a integração e a troca de dados entre as diferentes camadas da aplicação.

Neste projeto, foi desenvolvida uma API RESTful, responsável por fazer a ponte entre o Front-End (aplicativo mobile) e o Back-End (banco de dados e servidor). A API segue o padrão REST (Representational State Transfer), que utiliza os métodos HTTP — GET, POST, PUT e DELETE — para realizar as operações de leitura, criação, atualização e exclusão de dados.

Os dados são transmitidos no formato JSON (JavaScript Object Notation), facilitando a manipulação e interpretação das informações entre as partes do projeto. Essa estrutura garante maior flexibilidade e escalabilidade ao sistema, além de tornar o desenvolvimento mais ágil e padronizado.

Na figura 2, é um modelo de aplicação HTTP com as rotas determinadas do projeto para acesso do usuário.

Figura 2 - API Rotas com métodos HTTP



```
BackEndMobile > rotas > routers.js > ...
1  const express = require('express');
2  const router = express.Router();
3  const controllerSenai = require('../controllers/controllerSenai');
4
5  router.post('/cadastrar', controllerSenai.cadastrarUsuario); // cadastro de usuario
6  router.post('/login', controllerSenai.loginUsuario); // login
7
8  router.post('/cadastrarHabitto', controllerSenai.cadastrarHabitto); // cadastro habitto
9  router.get('/habitto/:usuario_id', controllerSenai.listarHabittoPorIDUsuario); // listar habitto por usuario id
10 router.get('/habitto', controllerSenai.listarTodosHabitto); // listar todos os habitto
11 router.get('/habittoID/:id', controllerSenai.listarHabittoID); // listar habitto por id
12
```

Fonte: A autora

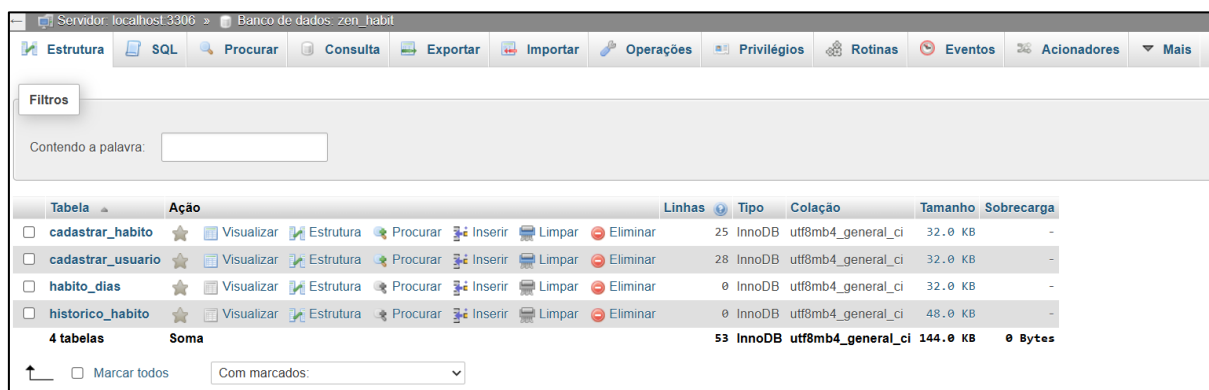
5.4 Banco de Dados

De acordo com a documentação oficial da Oracle (2023), “MySQL é o sistema de gerenciamento de banco de dados open source mais popular do mundo, oferecendo confiabilidade, desempenho e escalabilidade para aplicações web críticas” (MySQL Documentation, 2023).

No devido projeto, utilizou-se o MariaDB, derivado do MySQL, que garante compatibilidade, segurança e eficiência no gerenciamento dos dados. Ambos utilizam SQL para operações de inserção, atualização, exclusão e consulta de informações.

O MariaDB, por ser de código aberto, possui alta escalabilidade, robustez e ampla capacidade de armazenamento, ademais ele facilita a integração com linguagens como PHP e Python. A Figura 3 ilustra um exemplo da organização do banco de dados.

Figura 3 - Banco de Dados



The screenshot shows a web-based database management tool interface. At the top, there's a navigation bar with tabs like 'Estrutura', 'SQL', 'Procurar', 'Consulta', 'Exportar', 'Importar', 'Operações', 'Privilegios', 'Rotinas', 'Eventos', 'Acionadores', and 'Mais'. Below this is a search bar labeled 'Filtros' with the text 'Contendo a palavra:'. The main area displays a table structure for 'Banco de dados: zen_habit'. The table has columns: 'Tabela', 'Ação', 'Linhas', 'Tipo', 'Colaço', 'Tamanho', and 'Sobrecarga'. It lists four tables: 'cadastrar_habito', 'cadastrar_usuario', 'habito_dias', and 'historico_habito'. Each table row includes icons for actions like 'Visualizar', 'Estrutura', 'Procurar', 'Inserir', 'Limpar', and 'Eliminar'. At the bottom, there's a summary row for '4 tabelas' showing a total of 53 lines, InnoDB type, utf8mb4_general_ci collation, 144.0 KB size, and 0 Bytes of overhead.

Tabela	Ação	Linhas	Tipo	Colaço	Tamanho	Sobrecarga
cadastrar_habito	Visualizar, Estrutura, Procurar, Inserir, Limpar, Eliminar	25	InnoDB	utf8mb4_general_ci	32.0 KB	-
cadastrar_usuario	Visualizar, Estrutura, Procurar, Inserir, Limpar, Eliminar	28	InnoDB	utf8mb4_general_ci	32.0 KB	-
habito_dias	Visualizar, Estrutura, Procurar, Inserir, Limpar, Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
historico_habito	Visualizar, Estrutura, Procurar, Inserir, Limpar, Eliminar	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
4 tabelas	Soma	53	InnoDB	utf8mb4_general_ci	144.0 KB	0 Bytes

Fonte: A autora

5.6 JavaScript

O JavaScript é uma linguagem de programação interpretada vastamente utilizada em navegadores da web. De acordo com a documentação oficial da Mozilla (2023), o uso de recursos assíncronos, como Promises e Async/Await, permite que as aplicações mantenham a responsividade ao realizar tarefas em segundo plano, garantindo uma experiência de uso mais fluida e dinâmica.

A linguagem disponibiliza diversas ferramentas que possibilitam a criação de páginas web interativas, permitindo a modificação e integração em tempo real do conteúdo e da aparência dos elementos que compõem um documento.

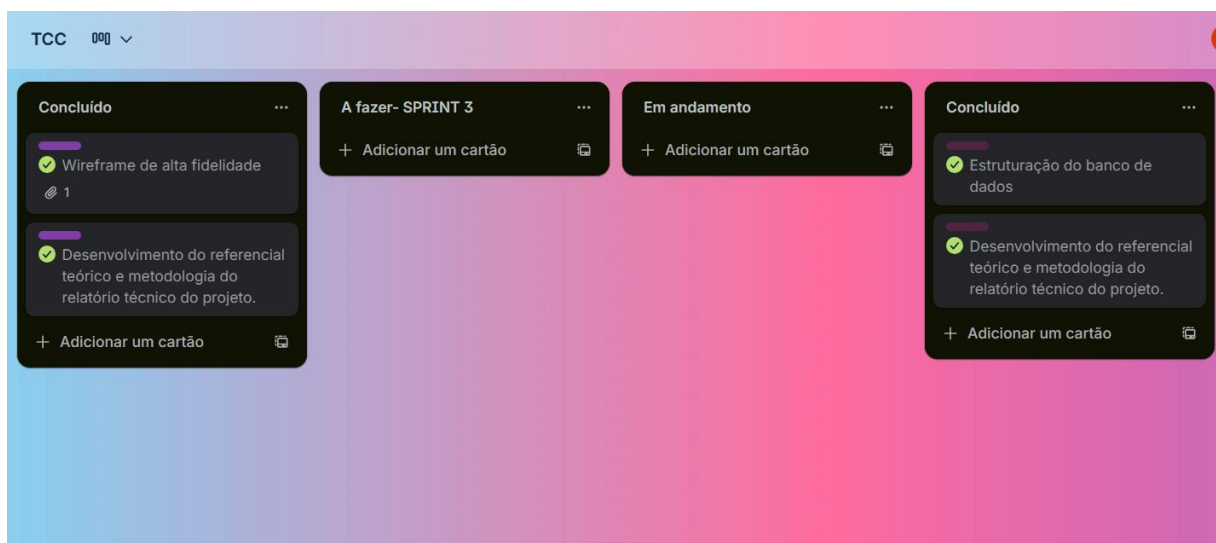
Além disso, o JavaScript é amplamente disposto em frameworks modernos, como React, Angular e Vue.js, que otimizam o desenvolvimento de aplicações web escaláveis, interativas e integradas, garantindo maior eficiência e flexibilidade no decorrer do processo de programação.

5.7 Trello

De acordo com Mello e Souza (2020), o Trello se destaca como uma ferramenta de gestão de projetos devido à sua interface intuitiva e ao suporte integrado à metodologia Scrum. A plataforma baseia-se no método Kanban, proporcionando uma visualização clara e organizada das fases de um projeto.

Além disso, contribui para o monitoramento das etapas de execução e facilita a interação entre os setores, promovendo uma gestão mais eficiente das informações e das tarefas. Na Figura 4, apresenta-se um exemplo do Trello utilizado durante o desenvolvimento do projeto.

Figura 4 - Trello



Fonte: A autora

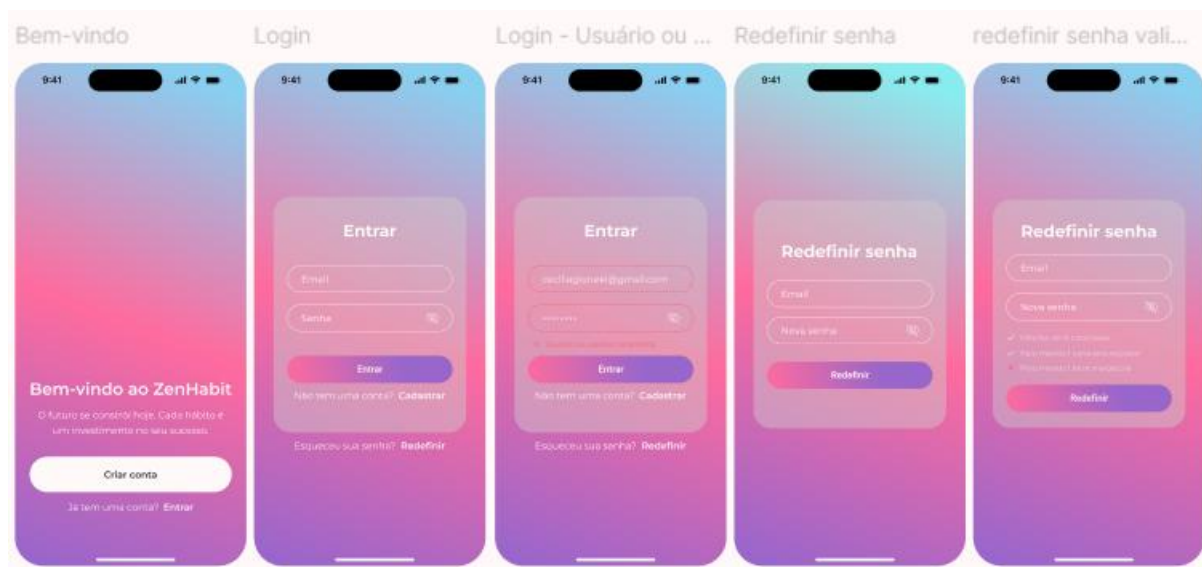
5.8 Figma

O Figma é uma das plataformas mais utilizadas por profissionais de prototipagem, design de interface de usuário (UI) e experiência do usuário (UX). Segundo Dustin Tanner, Líder de Sistemas de Design do Cash App, “O Figma é a única ferramenta onde você pode fazer tudo em um só lugar, e essa centralização é fundamental.”

A ferramenta permite integrar design, prototipagem e colaboração em tempo real, facilitando ajustes e melhorias de maneira ágil. Além disso, oferece recursos visuais e interativos que contribuíram para a construção de layouts intuitivos, garantindo uma interface clara e eficiente para o aplicativo.

Na Figura 5, apresenta-se um exemplo do Figma utilizado como referência durante a criação do projeto.

Figura 5 - Figma



Fonte: A autora

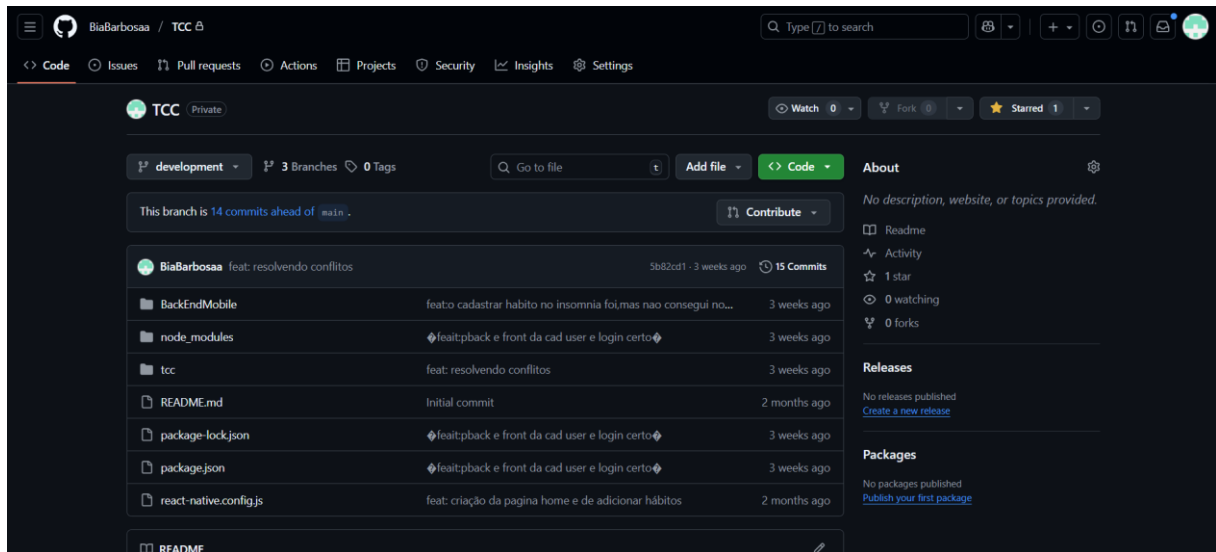
5.9 Git e GitHub

O GitHub foi utilizado como principal plataforma de versionamento e colaboração ao longo do projeto. Conforme estudo apresentado na ACM Conference on Computer-Supported Cooperative Work (Santos et al., 2021), a ferramenta não apenas facilita o controle de versões, mas também promove um ambiente colaborativo eficiente entre desenvolvedores, reduzindo problemas de sincronização em equipes distribuídas.

Além disso, o GitHub atua como um repositório essencial para o gerenciamento do código-fonte e das versões desenvolvidas, garantindo organização e rastreabilidade do projeto. O Git Bash foi empregado para executar comandos de atualização de pacotes do Node.js e rodar o servidor local, permitindo a visualização do aplicativo no navegador.

Na Figura 6, apresenta-se um exemplo do GitHub utilizado durante o desenvolvimento do projeto.

Figura 6 – Git e GitHub



Fonte: A autora

5.10 Insomnia

Conforme a documentação oficial do Insomnia (2025), a plataforma oferece um ambiente colaborativo voltado para o design, teste e depuração de APIs, auxiliando desenvolvedores na criação de interfaces mais eficientes e organizadas. Sua interface intuitiva e a possibilidade de estruturar requisições em pastas facilitam o gerenciamento e a execução de testes em tempo real.

O Insomnia foi utilizado neste projeto para validar o funcionamento das rotas da API, garantindo que as requisições e respostas fossem testadas a fim de assegurar o seu funcionamento conforme o esperado. Sua praticidade e recursos visuais contribuíram significativamente para o monitoramento e a otimização do back-end durante o desenvolvimento da aplicação.

6 RESULTADOS

O desenvolvimento do aplicativo para a criação e o monitoramento de hábitos teve como foco a organização pessoal e a manutenção de práticas voltadas à saúde. Ao integrar funcionalidades como a criação, edição e gerenciamento de hábitos, foi possível otimizar a experiência do usuário, promovendo maior organização, constância e comprometimento no cumprimento das metas diárias.

Para a construção do software, foram utilizadas tecnologias modernas como o React Native, que permitiu desenvolver uma interface responsiva e intuitiva, e o banco de dados MySQL, que garantiu um armazenamento estruturado e confiável das informações. A implementação de uma API em Node.js possibilitou a comunicação eficaz entre o front-end e o back-end, aprimorando o desempenho geral da aplicação.

Por se tratar de um projeto desenvolvido individualmente, a organização e o planejamento tiveram papel essencial para o sucesso do aplicativo. A divisão das etapas de forma estruturada — desde a definição das funcionalidades até o desenvolvimento e os testes — permitiu conduzir o trabalho com clareza e eficiência. Esse processo favoreceu o aprimoramento das habilidades técnicas e evidenciou a importância da autonomia no desenvolvimento de soluções tecnológicas completas.

7 CONCLUSÃO

A iniciativa Zen Habit evidenciou o impacto positivo que a união entre tecnologia e organização pessoal pode proporcionar no gerenciamento e acompanhamento de hábitos. Por meio de uma aplicação móvel moderna e de fácil acesso, tornou-se possível simplificar o processo de criação e monitoramento de hábitos, oferecendo aos usuários uma ferramenta prática, intuitiva e alinhada às demandas atuais de quem busca mais disciplina e qualidade de vida.

A inclusão de recursos como ícones ilustrativos e uma interface visualmente clara contribuiu para tornar a experiência mais motivadora e personalizada. Essas funcionalidades reforçaram a proposta do aplicativo ao incentivar a constância, facilitar o uso diário e promover uma rotina mais equilibrada. Dessa forma, o Zen Habit se consolidou como um apoio direto ao desenvolvimento pessoal e ao autocuidado.

O desenvolvimento do aplicativo também demonstrou como soluções tecnológicas simples e bem estruturadas podem impactar positivamente o cotidiano das pessoas. O projeto ressaltou a importância da autonomia, do planejamento e da aplicação prática dos conhecimentos adquiridos ao longo da formação. Assim, o Zen Habit não apenas cumpriu seu objetivo funcional, como também evidenciou o potencial transformador da tecnologia na criação de ferramentas voltadas ao bem-estar e à melhoria dos hábitos diários.

8.REFERÊNCIAS

DJANGO SOFTWARE FOUNDATION. **Django documentation: Overview**. 2024. Disponível em: <https://docs.djangoproject.com/en/5.0/intro/overview/>. Acesso em: 16 set 2025.

HOSPITAL ISRAELITA ALBERT EINSTEIN. **Hábitos saudáveis: o guia para construir os seus com a família**. Disponível em: <https://vidasaudavel.einstein.br/habitos-saudaveis/>. Acesso em: 12 ago. 2025.

INSOMNIA. **Insomnia: A plataforma colaborativa de desenvolvimento de APIs**. Disponível em: <https://insomnia.rest/>. Acesso em: 6 nov. 2025.

MICROSOFT. **Visual Studio Code Documentation**. 2023. Disponível em: <https://code.visualstudio.com/docs>. Acesso em: 16 set 2025.

MOZILLA. **Async/Await. MDN Web Docs**, 2023. Disponível em: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Asynchronous/Async_await. Acesso em: 16 set 2025.

NODE.JS. **About Node.js**. 2023. Disponível em: <https://nodejs.org/en/about>. Acesso em: 16 abr 2025.

ORACLE. **MySQL 8.0 Reference Manual**. 2023. Disponível em: <https://dev.mysql.com/doc/refman/8.0/en/>. Acesso em: 11 abr 2025.

SANTOS, R. et al. **Collaborative Coding Practices in Distributed Teams**. In: ACM CSCW, 2021. Disponível em: <https://dl.acm.org/doi/10.1145/3442381.3449852>. Acesso em: 16 maio 2025.

MELLO, Anna Carolina; SOUZA, Luiz Henrique Gomes de. **Solução Simplificada para o Monitoramento e Controle de Projetos Utilizando a Ferramenta Trello**. Revista Boletim do Gerenciamento, v. 2, n. 2, out. 2018. Disponível em: <https://nppg.org.br/revistas/boletimdogerenciamento/article/view/35>. Acesso em: 28 out. 2025.

META. **React Native – Learn once, write anywhere**. Disponível em: <https://reactnative.dev/>. Acesso em: 6 nov. 2025

OECD. **The impact of digital technologies on well-being: Main insights from the literature. OECD Papers on Well-being and Inequalities**, No. 29, OECD Publishing, Paris, 2024. Disponível em: <https://doi.org/10.1787/cb173652-en>. Acesso em: 12 ago. 2025.

SOUSA, Eduardo Pereira Martins de. **Emulação de um gerenciador de dados orientado a objetos através de uma interface de programação de aplicativos sobre um gerenciador relacional**. 2000. 170 f. Dissertação (Mestrado em Ciências da Computação) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2000. Disponível em: https://www.teses.usp.br/teses/disponiveis/55/55134/tde-01072003-163859/publico/Sousa_Mestrado.pdf. Acesso em: 6 nov. 2025.

TANNER, Dustin. **FIGMA. Professional Plan: Pricing & Features.** Disponível em: <https://www.figma.com/professional/>. Acesso em: 6 nov. 2025

APÊNDICE A – BIBLIOTECAS UTILIZADAS NO PROJETO FRONT-END

```
"dependencies": {
  "@react-native-async-storage/async-storage": "^2.1.2",
  "@react-native-picker/picker": "^2.11.4",
  "@react-native-vector-icons/common": "^12.4.0",
  "@react-native-vector-icons/fontawesome6": "^12.3.0",
  "@react-native-vector-icons/ionicons": "^12.3.0",
  "@react-native/new-app-screen": "0.81.0",
  "@react-navigation/bottom-tabs": "^7.4.8",
  "@react-navigation/native": "^7.1.9",
  "@react-navigation/native-stack": "^7.3.13",
  "add": "^2.0.6",
  "auth": "^1.2.3",
  "axios": "^1.13.2",
  "material-ui-community-icons": "^0.15.0",
  "moment": "^2.30.1",
  "react": "19.1.0",
  "react-native": "0.81.0",
  "react-native-asset": "^2.1.1",
  "react-native-gesture-handler": "^2.25.0",
  "react-native-keyboard-aware-scroll-view": "^0.9.5",
  "react-native-keyboard-aware-scrollview": "^2.1.0",
  "react-native-linear-gradient": "^2.8.3",
  "react-native-modal": "^14.0.0-rc.1",
  "react-native-safe-area-context": "^5.5.2",
  "react-native-screens": "^4.10.0",
  "react-native-svg": "^15.12.1",
  "react-native-swiper": "^1.6.0",
  "react-native-vector-icons": "^10.3.0"
},
"devDependencies": {
  "@babel/core": "^7.25.2",
```

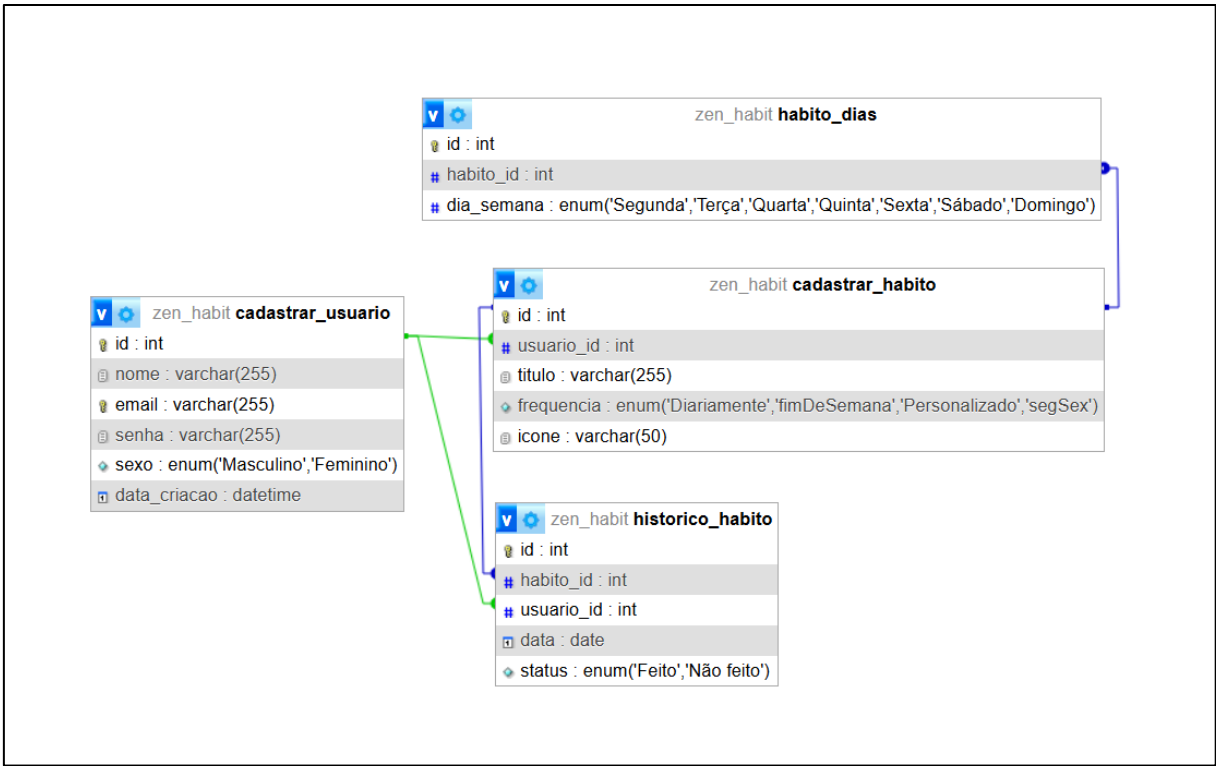
```
"@babel/preset-env": "^7.25.3",
"@babel/runtime": "^7.25.0",
"@react-native-community/cli": "20.0.0",
"@react-native-community/cli-platform-android": "20.0.0",
"@react-native-community/cli-platform-ios": "20.0.0",
"@react-native/babel-preset": "0.81.0",
"@react-native/eslint-config": "0.81.0",
"@react-native/metro-config": "0.81.0",
"@react-native/typescript-config": "0.81.0",
"@types/jest": "^29.5.13",
"@types/react": "^19.1.0",
"@types/react-test-renderer": "^19.1.0",
"eslint": "^8.19.0",
"jest": "^29.6.3",
"prettier": "2.8.8",
"react-test-renderer": "19.1.0",
"typescript": "^5.8.3"
},
```

APÊNDICE B – BIBLIOTECAS UTILIZADAS NO PROJETO BACK-END

```
"dependencies": {  
  "axios": "^1.5.1",  
  "bcrypt": "^5.1.1",  
  "cors": "^2.8.5",  
  "dotenv": "^17.2.3",  
  "express": "^4.18.2",  
  "jsonwebtoken": "^9.0.2",  
  "latest": "^0.2.0",  
  "mysql2": "^3.6.1",  
  "nodemon": "^3.1.10"  
},  
"devDependencies": {  
  "jest": "^30.1.3",  
  "supertest": "^7.1.4"  
}  
}
```

APÊNDICE C – DIAGRAMA DO BANCO DE DADOS

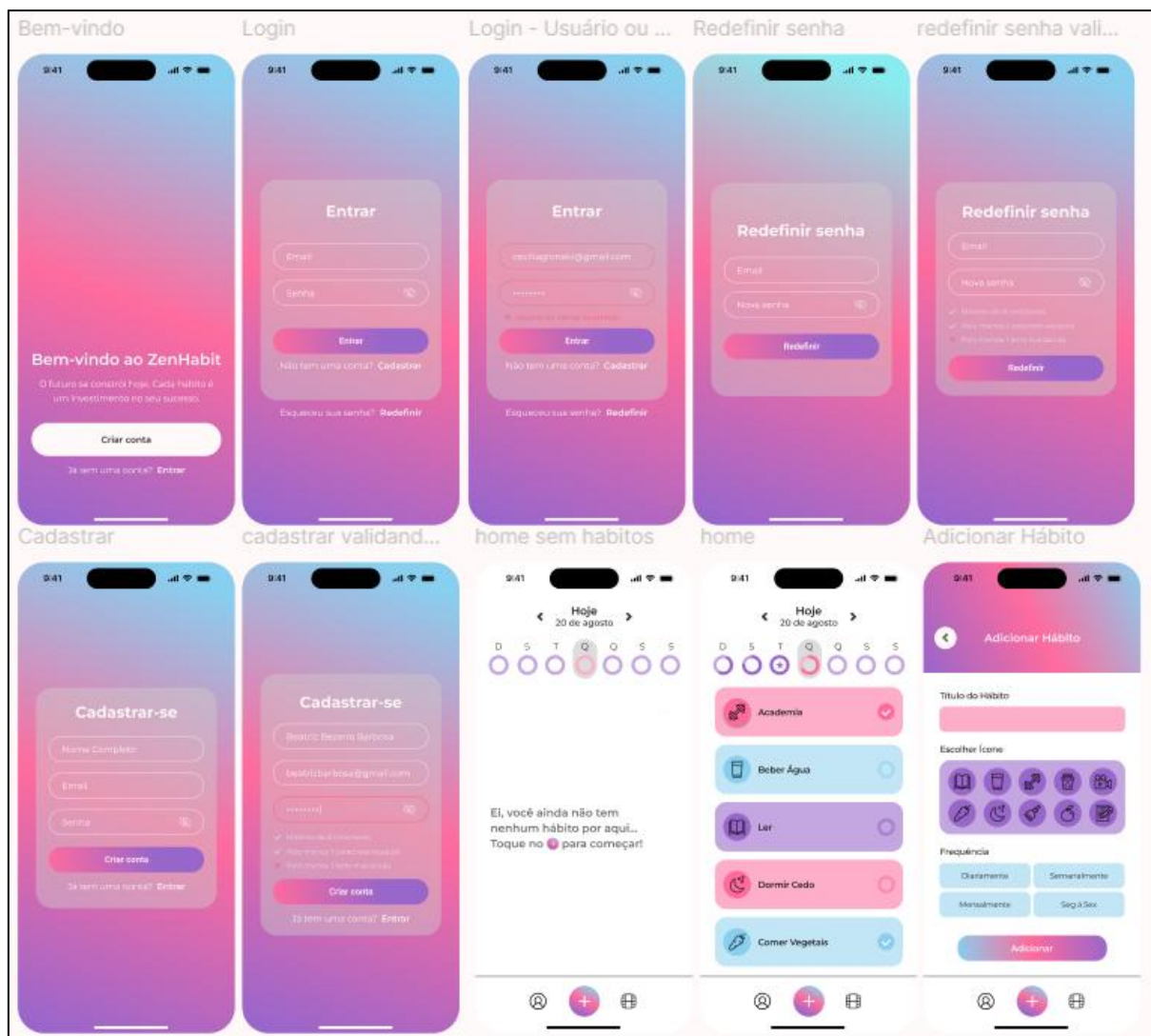
Figura 7 - Diagrama do Banco de Dados



Fonte: A autora

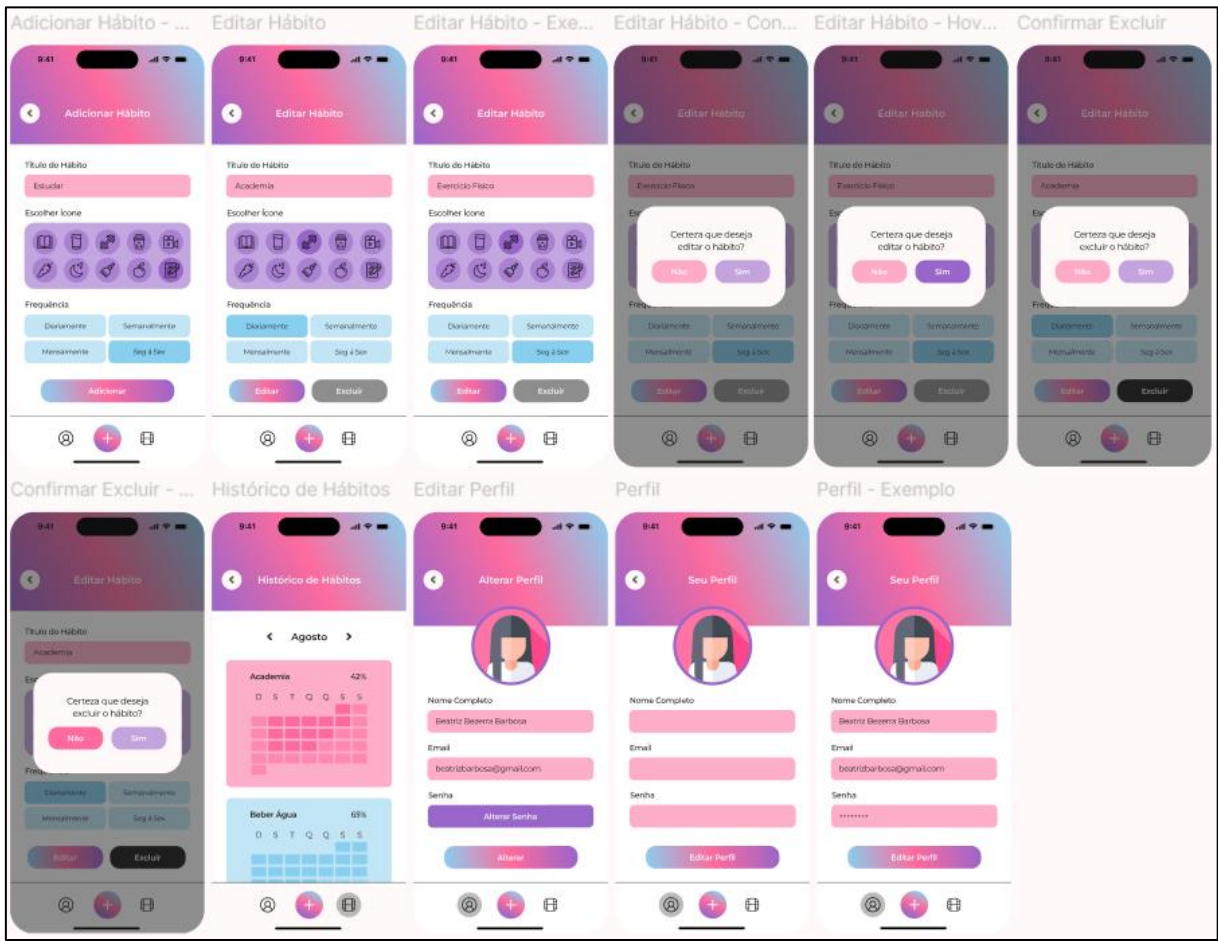
APÊNDICE D – PROTOTIPAÇÃO DAS TELAS DO APP

Figura 8 – Prototipação do Figma



Fonte: A autora

Figura 9 - Prototipação do Figma



Fonte: A autora