

### PUC MINAS (PUCMG)

Campus Liberdade

Graduação em Engenharia de Software

MIGUEL ALVES FERNANDES MORAES

MELYSSA MEIRELES DOS SANTOS

GABRIEL HEROLT ARAÚJO

**Software para auxiliar jovens a diferenciar cursos de Tecnologia da Informação da graduação**

TIAW – Engenharia de Software

Orientadores: João Luiz Silva Barbosa

Simone Alves Nogueira

Belo Horizonte - MG

2022

Sumário

[PUC MINAS (PUCMG) 1](#_Toc107954016)

[**1.** Introdução 3](#_Toc107954017)

[1.1 - Problemas 3](#_Toc107954018)

[1.2 - Objetivos 3](#_Toc107954019)

[1.3- Justificativa 4](#_Toc107954020)

[1.4 - Público-Alvo 4](#_Toc107954021)

[**2.** Especificações do Projeto 5](#_Toc107954022)

[2.1 – Personas e Mapas de Empatia 5](#_Toc107954023)

[2.2 – Histórias de Usuários 6](#_Toc107954024)

[2.3 – Requisitos 7](#_Toc107954025)

[2.3.1 – Requisitos Funcionais 7](#_Toc107954026)

[2.3.2 – Requisitos Não-Funcionais 7](#_Toc107954027)

[2.3 – Restrições 8](#_Toc107954028)

[**3.** Projeto de Interface 9](#_Toc107954029)

[3.1 – User Flow 10](#_Toc107954030)

[3.2 – Wireframes 12](#_Toc107954031)

[3.2.1 – Tela inicial: 12](#_Toc107954032)

[3.2.2 – Notícias: 13](#_Toc107954033)

[3.2.3 – Cursos: 14](#_Toc107954034)

[3.2.4 – Curso: 14](#_Toc107954035)

[3.2.5 – Universidades: 15](#_Toc107954036)

[3.2.6 – Formulário: 16](#_Toc107954037)

[3.2.7 – Resultado do formulário: 16](#_Toc107954038)

[**4.** Metodologia 18](#_Toc107954039)

[4.1 – Divisão de papeis 18](#_Toc107954040)

[4.2 – Ferramentas 18](#_Toc107954041)

[4.2 – Controle de versão 19](#_Toc107954042)

[**5.** Projeto da Solução 19](#_Toc107954043)

[5.1 – Tecnologias ultilizadas 19](#_Toc107954044)

[5.2 – Arquitetura da solução 21](#_Toc107954045)

[**6.** Planos de Teste 22](#_Toc107954046)

[6.1 – Testes 22](#_Toc107954047)

[6.1.1 – Acessar rota de administrador 22](#_Toc107954048)

[6.1.2 – Cadastrar e visualizar manchete 24](#_Toc107954049)

[6.1.4 – Cadastrar e visualizar curso 25](#_Toc107954050)

[6.1.5 – Realizar Formulário 26](#_Toc107954051)

[**7.** Referências bibliográficas 27](#_Toc107954052)

# Introdução

Este projeto aborda a dificuldade que os jovens encontram ao escolher o curso de tecnologia da informação que desejam cursar na graduação.

## 1.1 - Problemas

Aproximando-se do período de escolha de graduações, os joves que se interessam por tecnologia da informação passam por um impasse. Existe-se uma grande dificuldade em entender e diferenciar os tópicos abordados nas graduações, gerada por má informação de toda a comunidade e a falta de organização nos conteúdos explicativos disponibilizados na internet.

Além disso, fala-se muito dos cursos “Ciências da Computação” e “Engenharia da Computação”, ofuscando os demais cursos. Por estes motivos, os jovens podem acabar optando por um curso com o qual não se identificam tanto.

## 1.2 - Objetivos

Tem-se como objetivo geral desenvolver um software que atua como teste vocacional, auxiliando jovens a entenderem as áreas de TI. Tal teste visa direcionar os usuários para o curso de graduação que mais combina com seu perfil.

Para isso, adotou-se uma estratégia que consiste em fazer perguntas ao usuário voltadas à área de tecnologia da informação, com o intuito de identificar o seu perfil. Uma vez analisado, pode-se assumir quais atividades o usuário mais se adequa, apresentando-o ao curso que suprirá as expectativas do usuário relacionadas ao conteúdo abordado na graduação.

## 1.3- Justificativa

Por conta de os integrantes deste trabalho terem passado por dificuldades em encontrarem o curso de TI ideal, motivou-se o desenvolvimento de uma solução para este problema. Após validar que este é um problema real, por meio do processo de Empatia, explicado posteriormente neste documento, trabalhou-se em uma aplicação com o objetivo de auxiliar jovens a entender as diferenças entre as áreas de TI e também a encontrarem qual o seu curso ideal.

## 1.4 - Público-Alvo

Para identificar o nosso público alvo, fizemos entrevistas qualitativas com jovens que se interessam pela área de tecnologia e pretendem cursar a graduação. Verificou-se que estes indivíduos passam pelos mesmos problemas listados, e, com isso, embasou-se neles para a definição de nosso público-alvo.

Chegou-se a conclusão que os principais alvos são pessoas jovens, entre 14 e 25 anos, que querem aprender sobre tecnologia e demonstraram interesse em cursar a graduação, ou já cursam o ensino superior e não se identificaram com o tema escolhido. Ao se deparar com a dificuldade de definir qual o melhor curso de tecnologia, pessoas com este perfil serão potenciais usuários desta aplicação.

# Especificações do Projeto

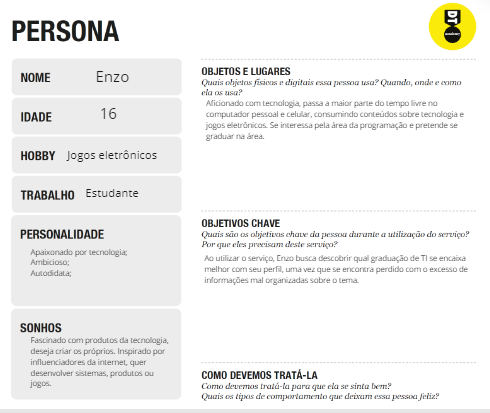
Nesta parte do projeto, utilizou-se de várias ferramentas e técnicas para auxiliar em sua produção. Entre elas estão:

* Trello, para auxiliar a produção e manutenção do quadro *Kanban*. (Trello, 2022).
* Miro, para a produção de mapas mentais, wireframes e o processo de Design Thinking (Miro, 2022).
* Git, para o versionamento de código (Git-scm, 2017).
* GitHub, como repositório na nuvem do projeto (GitHub, 2022).
* Design Thinking, para a captação de ideias para o projeto (Rock Content, 2019).
* Scrum, como metodologia ágil adotada para melhor produtividade (Britrix24, 2021)

## 2.1 – Personas e Mapas de Empatia

Na criação de personas, tomou-se como base o perfil médio dos entrevistados que demonstraram interesse no projeto. As características marcantes deste perfil são: Jovens, com interesse em tecnologia e que passam parte do tempo livre no computador.

**Figura 1-** Primeira persona



Fonte: Próprio autor

A persona criada é idealizada como o principal cliente do nosso sistema.

## 2.2 – Histórias de Usuários

|  |  |  |
| --- | --- | --- |
| Eu como persona | Quero/Preciso | Para |
| Enzo | Encontrar qual curso mais me agrada | Não cursar algo que não irá me agregar |
| Enzo | Encontrar faculdades próximas | Pois são mais acessíveis para mim |
| Enzo | Encontrar faculdades com preços que cabem no bolso | Pois não é fácil encontrar estes valores na internet |
| Enzo | Preciso saber o que é ofertado em cada curso da graduação | Para não acabar em um curso que me oferte coisas que não me interessam |

## 2.3 – Requisitos

As tabelas que a seguir apresentam os requisitos funcionais e não funcionais que detalham o escopo do projeto.

### 2.3.1 – Requisitos Funcionais

|  |  |  |
| --- | --- | --- |
| ID | Descrição do Requisito | Prioridade |
| RF-001 | Cadastrar usuário | Alta |
| RF-002 | Autenticar usuário | Alta |
| RF-003 | Alterar usuário | Baixa |
| RF-004 | Formulário para identificar o curso que o usuário mais se assemelha | Alta |
| RF-005 | Histórico de formulários respondidos pelo usuário | Média |
| RF-006 | Permitir que o usuário destaque cursos | Baixa |
| RF-007 | Exibir manchetes de cursos favoritos | Baixa |
| RF-008 | Exibir manchetes do tema “Cursos de TI” | Alta |
| RF-009 | Exibir universidades que possuem cursos de TI | Alta |
| RF-010 | Permitir a filtragem de universidades por preço | Media |
| RF-011 | Permitir a filtragem de universidade por curso | ALTA |
| RF-012 | Comparação de preços de faculdades particulares | Baixa |
| RF-013 | Listagem dos cursos de TI com uma breve descrição | Alta |

### 2.3.2 – Requisitos Não-Funcionais

|  |  |  |
| --- | --- | --- |
| ID | Descrição do Requisito | Prioridade |
| RNF-001 | O sistema deve ser responsivo para rodar em dispositivos móveis | Média |
| RNF-002 | Deve processar requisições do usuário em no máximo 3s | Baixa |
| RNF-003 | Base de dados que relaciona curso com perfil do usuário | Alta |

## 2.3 – Restrições

O projeto está restrito pelos itens apresentados na tabela a seguir

|  |  |
| --- | --- |
| ID | Restrição |
| 01 | O projeto deverá ser entregue até o final do semestre |
| 02 | Não pode ser desenvolvido um módulo de backend |

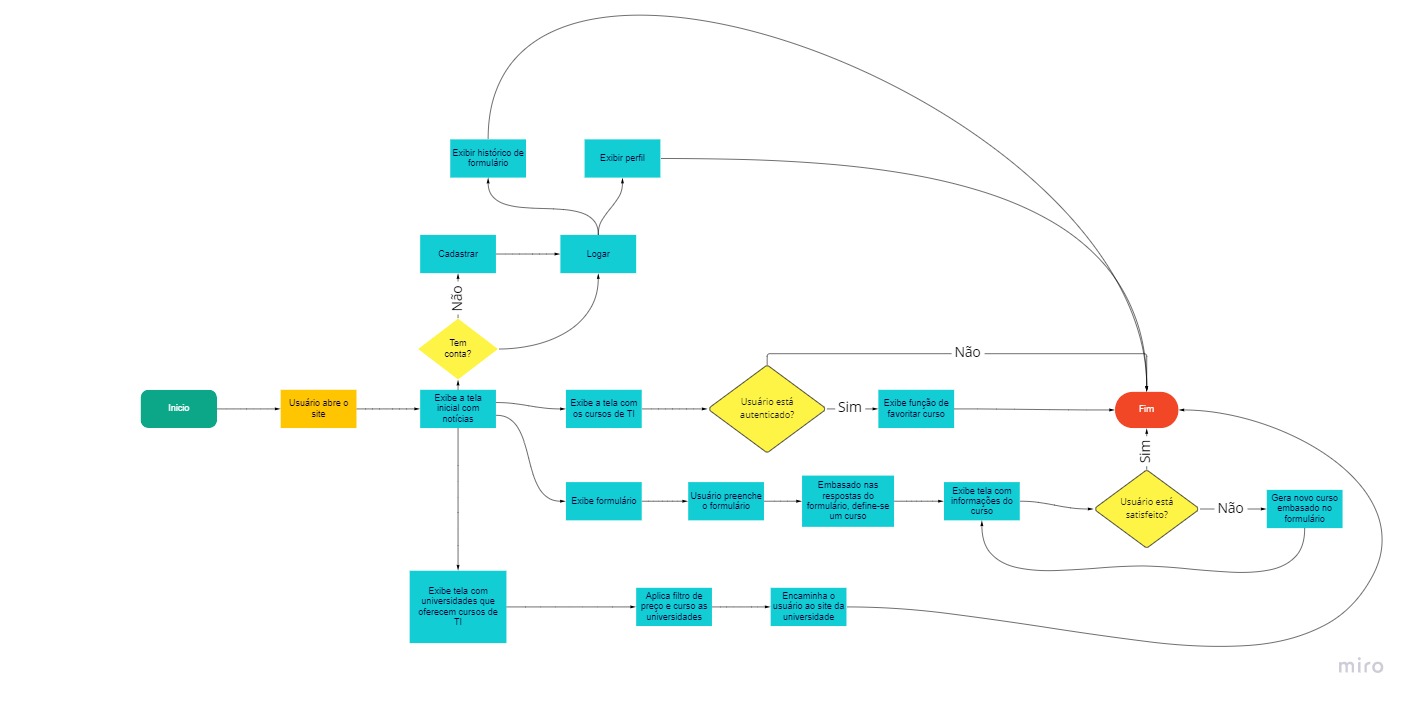
# Projeto de Interface

O projeto de interface fora feito utilizando a ferramenta de planejamento de projeto Miro. Nesta etapa, busca-se atender os requisitos funcionais e não funcionais especificados na seção 2.3, utilizando-se dos dados coletados nas pesquisas qualitativas. Para isso, fez-se útil o User Flow, mapeando a atividade do usuário dentro da aplicação. Também se fez uso dos Wireframes, que ajudam a dar uma pré-visualização da solução a ser desenvolvida.

Pensando na dificuldade do usuário em encontrar dados sobre cursos da área de TI na internet, fez-se uma prototipação focada em dar informações diretas e separadas por curso, além de uma ferramenta para auxiliar o usuário a definir qual curso melhor se encaixa em seu perfil.

## 3.1 – User Flow

**Figura 2-** User Flow principal

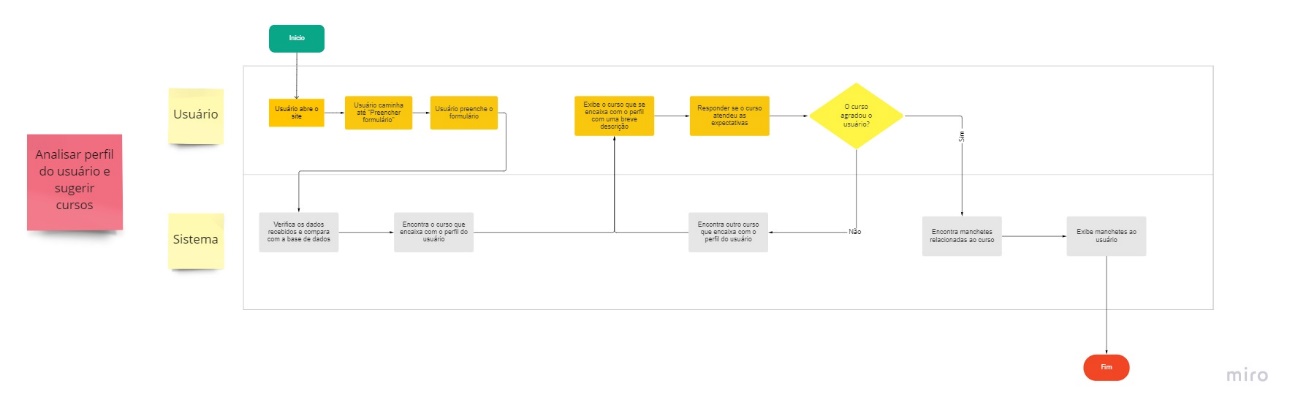


Fonte: Próprio autor

Pelo User Flow de toda aplicação é possível perceber que, ao acessar o site, permite-se quatro principais rotas: Cursos, universidades, formulário e autenticação.

No caminho de cursos, é possível ver uma listagem dos custos da área de TI e, se autenticando, acompanhar os que mais interessam ao usuário. Já no caminho de universidade, são listadas universidades que oferecem os cursos da área de TI, possibilitando o usuário filtrar por curso e preço. Em autenticação, engloba-se as funções de cadastro, edição e autenticação de usuário.

**Figura 3-** User Flow de Formulário



Fonte: Próprio autor

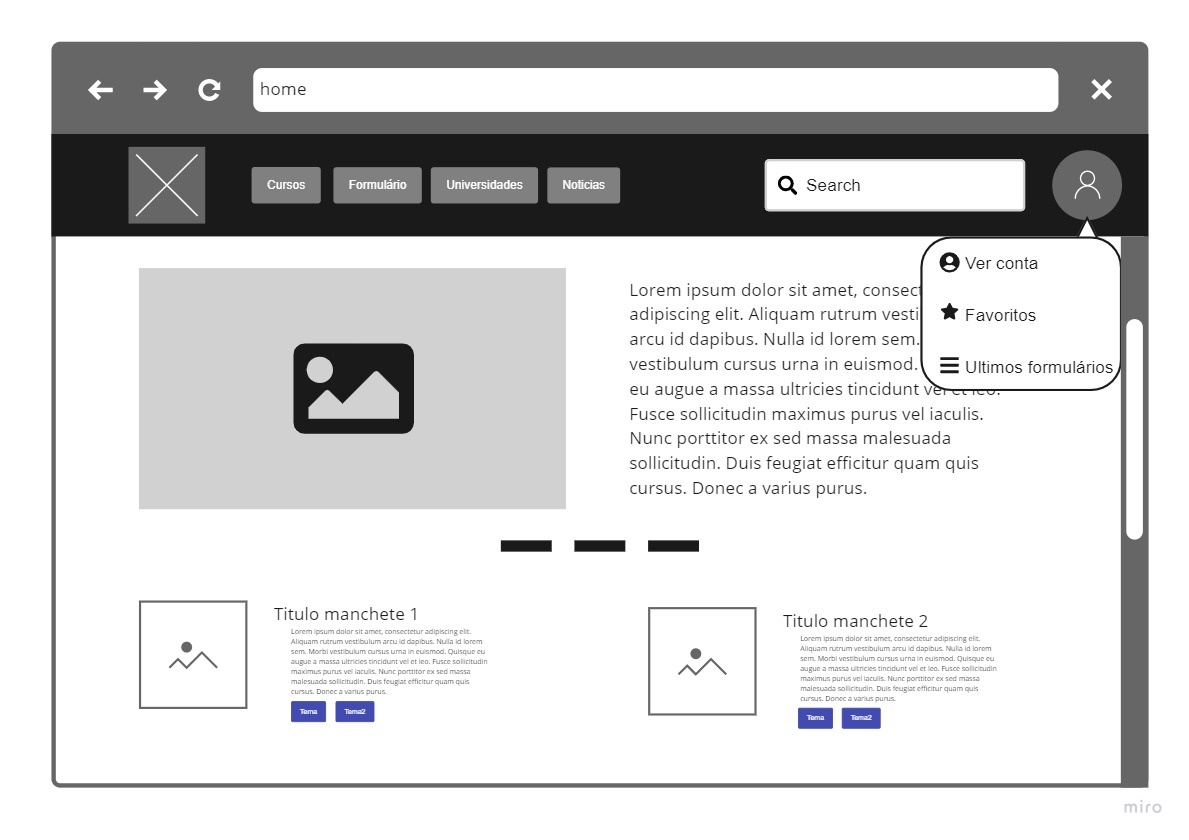
Por ser mais extenso, fora feito um *User Flow* apenas para o processo de formulário. Por ele, é possível verificar que, respondidas as perguntas solicitadas, o sistema, com base nas respostas, indicará ao usuário qual o curso que mais se assemelha ao seu perfil. Após obter o curso, exibirá manchetes e informações relevantes sobre. Depois da leitura, o usuário poderá informar ao sistema que aquele curso não o agrada, gerando-se um novo.

## 3.2 – Wireframes

Esta seção contém os *Wireframes* produzidos para esta aplicação.

### 3.2.1 – Tela inicial:

**Figura 4-** Wireframe da tela inicial

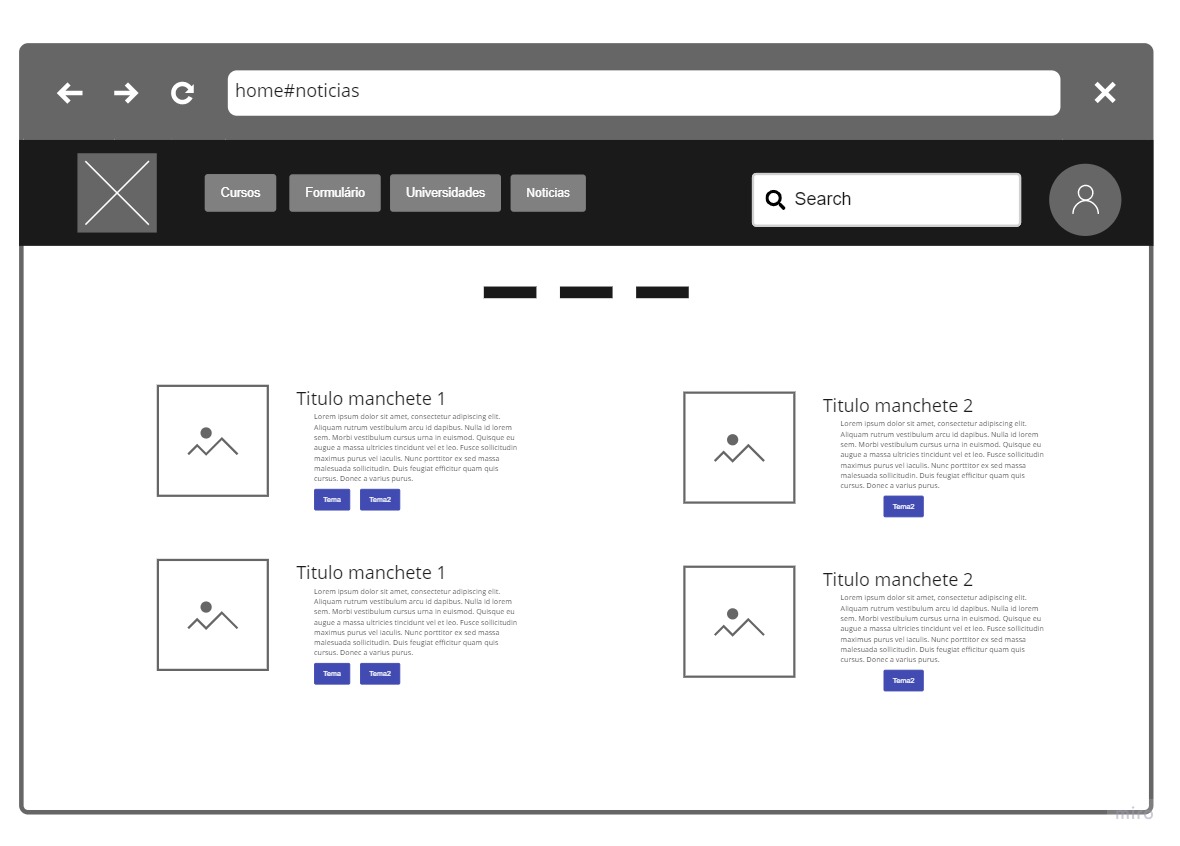


Fonte: Próprio autor

Esta tela é a tela inicial do projeto. Contém um header Navigation, com a função de navegar por toda aplicação. A tela inicial possuí um *Carrousel*, que destacará as principais notícias da área e manchetes sobre cursos de TI.

### 3.2.2 – Notícias:

**Figura 5-** Wireframe da tela de notícias

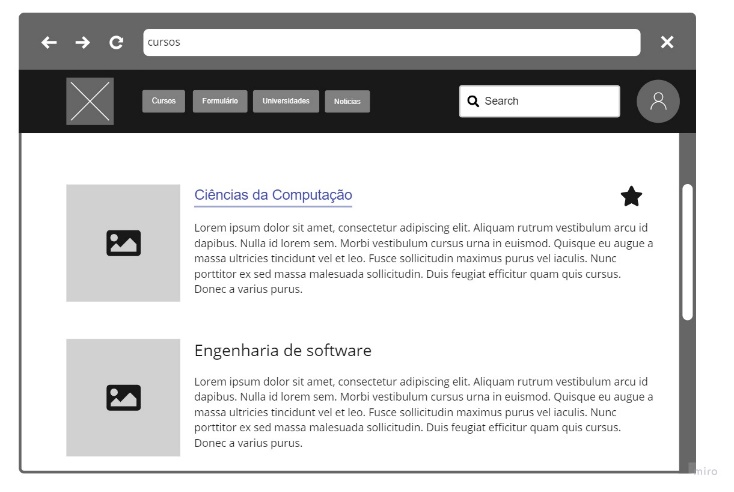


Fonte: Próprio autor

A tela de notícias ainda está dentro da tela inicial. Ao clicar no botão “Notícias” no header, o usuário será direcionado a seção de manchetes.

### 3.2.3 – Cursos:

**Figura 6-** Wireframe da tela de Cursos



Fonte: Próprio autor

Nesta tela, acessada pelo botão “Cursos” no header, será apresentado ao usuário uma lista de cursos com uma breve descrição. O usuário pode favoritar ou desfavoritar cursos.

### 3.2.4 – Curso:

**Figura 7-** Wireframe da tela de Curso

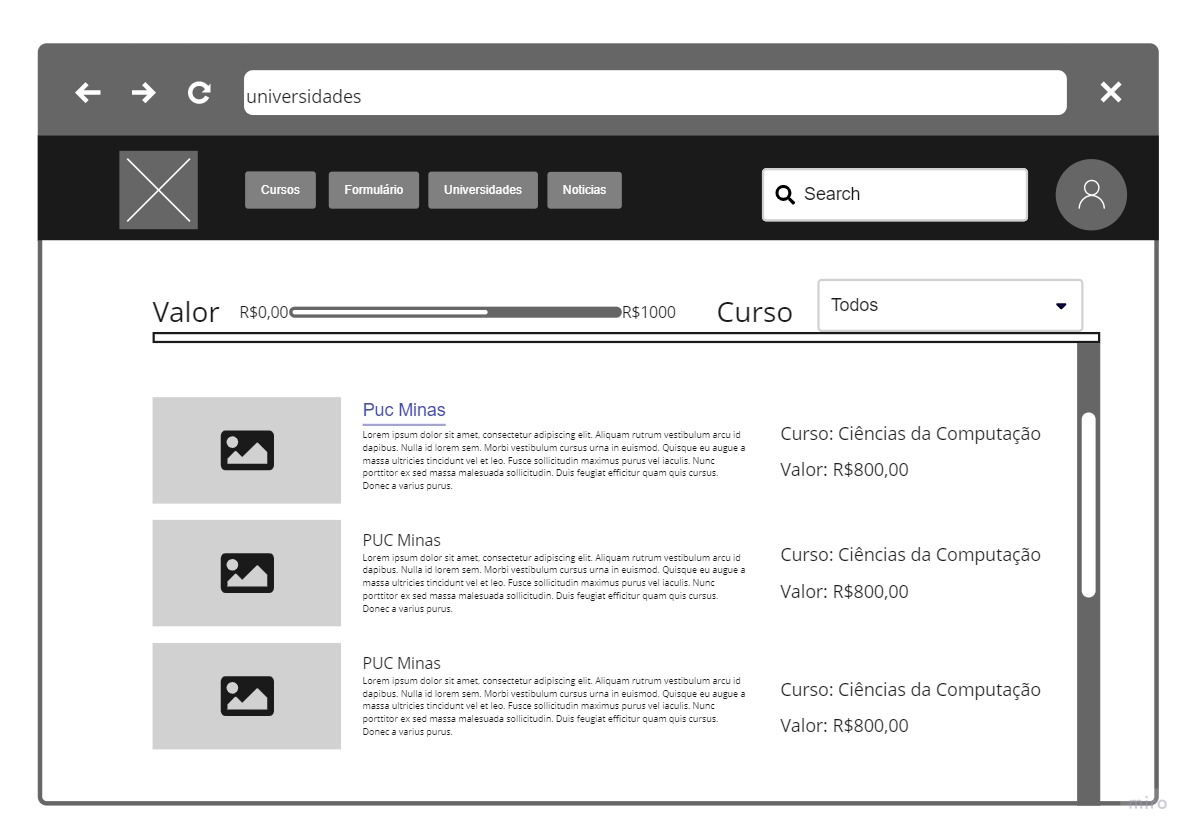


Fonte: Próprio autor

Ao selecionar um curso na tela de cursos, o usuário será redirecionado para uma nova tela, que detalhará melhor o curso escolhido e dará a opção de ver universidades que o oferecem.

### 3.2.5 – Universidades:

**Figura 8-** Wireframe da tela de Universidade

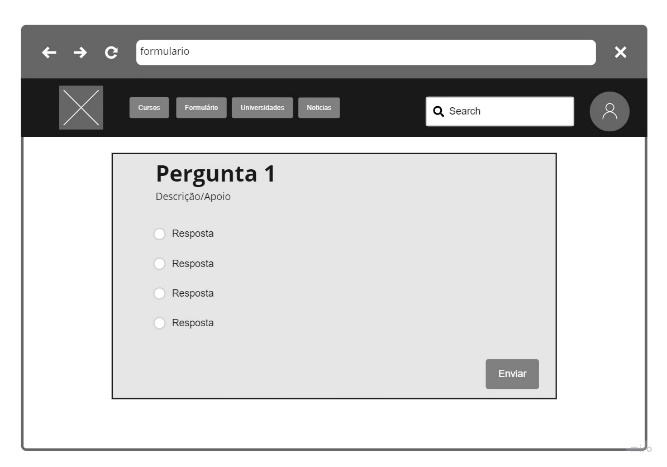


Fonte: Próprio autor

Clicando em “Universidades” no header, esta tela será exibida, listando todas universidades que oferecem cursos de TI. Também será possível filtrar as universidades exibidas por curso e preço. Ao clicar em uma universidade, o usuário será redirecionado para o site da instituição.

### 3.2.6 – Formulário:

**Figura 9-** Wireframe da tela de Formulário



Fonte: Próprio autor

Nesta página, também acessada pelo header, o usuário responderá perguntas e, com base nas respostas, o sistema encontrará um curso que se assemelha ao perfil do usuário.

### 3.2.7 – Resultado do formulário:

**Figura 8-** Wireframe da tela de Resultado



Fonte: Próprio autor

Ao preencher todo o formulário e clicar em enviar, o sistema definirá o curso que aparenta ser do gosto do usuário e o encaminhará para uma página com os dados do curso. Estes dados são: uma breve descrição e manchetes do curso ou universidades que o oferecem.

**Figura 9-** Wireframe da tela de Resultado 2



Fonte: Próprio autor

Clicando em “Me dê outro curso”, o sistema retornará outro curso que se assemelha ao perfil obtido no formulário.

# Metodologia

Antes da iniciação do desenvolvimento, utilizou-se o processo de *Design Thinking* para delimitar as necessidades dos usuários, criando-se uma persona principal para o nosso projeto. Uma vez criada, a persona facilita a prototipação do sistema, tornando-se palpável a criação de designs que agradam o usuário final.

Para a realização do trabalho, adotamos a metodologia *Scrum*. Para cada tarefa, designamos a dificuldade de execução, definindo-se um prazo relativo à dificuldade. Para cada funcionalidade, faz-se uma reunião com o consumidor final para validar a sua necessidade.

Para incrementar o *Scrum*, utilizou-se a metodologia ágil *Kanban*, para se ter uma melhor visualização das tarefas e seus atuais estados. Para a metodologia ágil Kanban, estamos utilizando a ferramenta *Trello.*

## 4.1 – Divisão de papeis

Para a divisão de papéis, utilizou-se a pontuação de dificuldade de cada tarefa definida pela metodologia *Scrum.* Dividiu-se as tarefas em grupos de pontuações similares, e, cada integrante do grupo, ficou responsável por um destes grupos de tarefas.

## 4.2 – Ferramentas

O editor de código escolhido foi o *VsCode*. Além de ter extensões compatíveis com HTML e CSS (Visual Studio Code, 2021), possuí também ferramentas para visualizar o controle de versão e o *Git Flow.*

Para comunicação, estamos utilizando o Trello. Por meio de um quadro *Kanban* as tarefas são divididas, permitindo que todos possam documentar o seu progresso nas tarefas.

Na diagramação, utilizamos o Miro. Por ser uma ferramenta que possibilita a criação de fluxogramas, wireframes e outros processos de prototipação, escolhemos o Miro a fim de salvar tempo e melhorar a produtividade, uma vez que dominando a ferramenta é possível realizar todas as etapas visuais de prototipação.

Como hospedagem, escolhemos o Heroku, que, além de ser gratuito, permite a conexão direta com o github sem muita burocracia.

|  |  |  |
| --- | --- | --- |
| Ambiente | Plataforma | Link de Acesso |
| Processo de Design Thinking | Miro | <https://miro.com/app/board/uXjVOA3Il1A=/> |
| Repositório de código | Github | https://github.com/ICEI-PUC-Minas-PPLES-TI/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti |
| Hospedagem de site | Heroku | <https://entedimento-das-areas-de-ti.herokuapp.com/> |
| Protótipo interativo | Miro | https://miro.com/app/board/uXjVO7xkYGI=/ |

## 4.2 – Controle de versão

Para o controle de versão, adotou-se o *Git*, permitindo que a equipe trabalhe, dentro do projeto, de forma síncrona. Para evitar conflitos durante a mescla de arquivos, adotou-se as ramificações, que permitem a criação de versões paralelas do projeto, facilitando a divisão de código que está em processo de testes e melhorias do que já está em produção.

# Projeto da Solução

## 5.1 – Tecnologias ultilizadas

O desenvolvimento do projeto ocorreu parcialmente no sistema operacional Ubuntu 18.04 e parcialmente em sua nova versão, o Ubuntu 20.10. O ambiente utilizado para a programação foi o Visual Studio Code, uma IDE (*Integrated Development Environment*) utilizada para edição de código-fonte.

Desenvolvido pela Microsoft para Windows, Linux e macOS e lançado em 2015, o Visual Studio Code é um software livre e de código aberto que inclui suporte para depuração e refatoração do código. Sendo um editor de código-fonte, torna-se possível programar em inúmeras linguagens, incluindo Dart.

Foram utilizados também os softwares Dia (DIA WEBSITE, 2020) e Miro (MIRO WEBSITE, 2020), aplicações gratuitas para desenho de diagramas.

Estarão listadas outras tecnologias e bibliotecas utilizadas para o desenvolvimento do sistema, assim como suas funções.

1. GitHub:  plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git (GITHUB WEBSITE, 2020), utilizada para versionamento do código do projeto;
2. Trello: Aplicação de gerenciamento de projeto que facilita a visualização de tarefas e seu estado de produção a partir de um quadro Kanban (TRELLO WEBSITE, 2020). Utilizado para organização, mantendo os orientadores informados sobre o andamento do projeto;
3. Kanban: Metodologia ágil[[1]](#footnote-1)e **visual adotada para o desenvolvimento do sistema, auxiliando no** controle de produção ou gestão de tarefas utilizando um sistema de colunas representando o estado de cada tarefa. As colunas são: A fazer, Em execução e Feito (ARTIA WEBSITE, 2019).
4. *Maps* *API*: *API* desenvolvida pelo Google que utiliza o *Google Maps* para exibição de um mapa local ou global (GOOGLE MAPS PLATFORM, 2019), utilizada para exibição do mapa dentro do aplicativo móvel*;*
5. JavaScript: Linguagem capaz de executar *scripts* do lado do usuário (DEVELOPER MOZILLA, 2020);
6. HTML: Linguagem de marcação utilizada para desenvolver o corpo de páginas *web*.
7. CSS: É uma folha de estilização de aparência em páginas *web* que utilizam HTML, XML e XHTML para o desenvolvimento (SILVA, 2008);
8. Bootstrap: Um framework web para desenvolvimento de *layouts* e telas de sistema *web*, facilitando o desenvolvimento do *front-end* através de classes CSS (WIKIPÉDIA, 2020), utilizado para estilização das interfaces de usuário do sistema *web* do projeto;
9. Data Tables: É *plug-in* para a biblioteca jQuery do JavaScript que permite a criação de tabelas flexíveis, com ferramentas de pesquisa, paginação, etc. (DATATABLES WEBSITE, 2019)*,* utilizado para estilização das interfaces de usuário do sistema *web* do projeto;
10. FontAwesome: Pacote de ícones para desenvolvimento de interfaces (FONTAWESOME WEBSITE, 2020) utilizado no módulo *web.*

## 5.2 – Arquitetura da solução

No projeto utilizou-se a distribuição de pastas no padrão *mvc* (Model, View, Control), separando os arquivos de visualização (*HTML* e *CSS*) dos arquivos que controlam quais informações serão exibidas (*JavaScript*). Para armazenamento de informações, utilizou-se o armazenamento local do navegador.

Para hospedagem, optou-se por duas ferramentas gratuitas. A decisão de utilizar de duas ferramentas se embasou nos testes futuros, que contabilizariam a performance de cada hospedagem. As hospedagens utilizadas foram o Github Pages, funcionalidade integrada ao *GitHub* que permite hospedar códigos de determinado repositório, e o *Heroku*.

# Planos de Teste

Nesta seção serão listados e detalhados planos de testes criados para o uso da aplicação. Cada plano contará com uma lista de etapas detalhadas e imagens explicativas.

## 6.1 – Testes

|  |  |  |
| --- | --- | --- |
| ID | Descrição | Página referente |
| Teste 01 | Acessar rota de administrador | Login |
| Teste 02 | Cadastrar e visualizar manchete | Manchete |
| Teste 03 | Cadastrar e visualizar curso | Curso |
| Teste 04 | Cadastrar e visualizar universidade | Universidade |
| Teste 05 | Realizar formulário | Formulario |

### 6.1.1 – Acessar rota de administrador

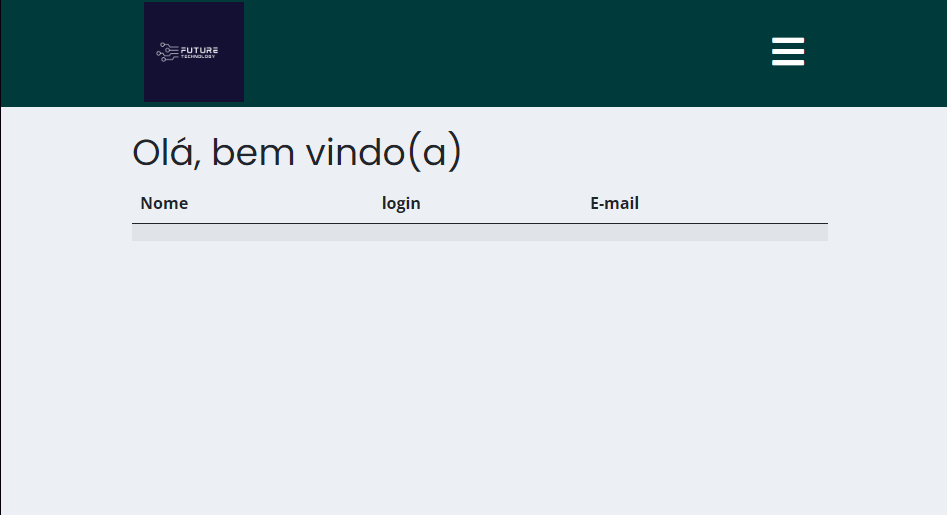
**Figura 10-** Tela de login



Fonte: Próprio autor

Acessando a *URL* <https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/login/login.html>, deve-se preencher os campos usuário e senha com os dados “admin” e “123”, respectivamente. Após preenchido, é necessário clicar em “entrar”. Recomenda-se utilizar dados aleatórios para validar a funcionalidade de entrar.

**Figura 11-** Tela inicial admin



Fonte: Próprio autor

Autenticado, o usuário deverá se deparar com a tela de boas-vindas, identificada pela *figura 11.* Caso ocorra, significará que o teste foi um sucesso.

### 6.1.2 – Cadastrar e visualizar manchete

**Figura 12-** Crud manchete



Fonte: Próprio autor

Autenticado como administrador e acessando a *URL* <https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/admin/crud-manchetes/manchetes.html>, deve-se deparar com a tela de gestão de manchetes. Nesta tela, é necessário selecionar “Criar nova manchete” e preencher os dados de nova manchete conforme solicitado.

Após cadastrado, deve-se acessar a tela inicial, endereçada pela URL <https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/index.html>, e verificar se a nova manchete se encontra na tela. Caso sim, indicará que o caso de teste foi um sucesso.

### 6.1.4 – Cadastrar e visualizar curso

**Figura 13-** Crud curso



Fonte: Próprio autor

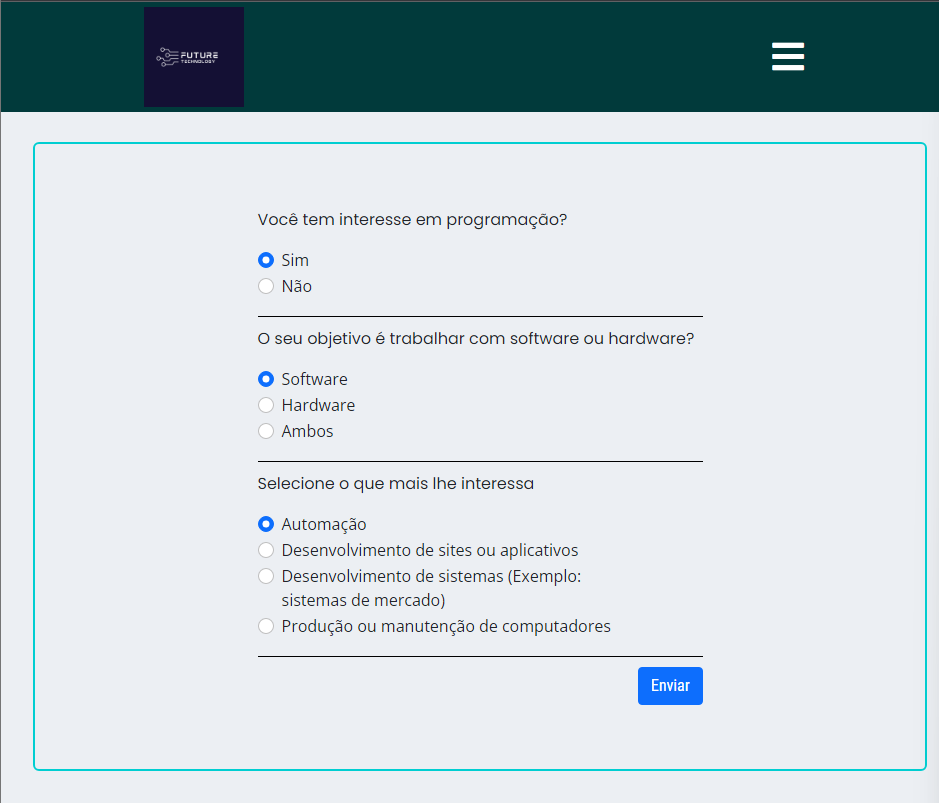
Autenticado como administrador e acessando a *URL* [*https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/admin/crud-cursos/cursos.html*](https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/admin/crud-cursos/cursos.html), deve-se deparar com a tela de gestão de cursos. Nesta tela, é necessário selecionar “Criar novo curso” e preencher os dados de novo curso conforme solicitado.

Após cadastrado, deve-se acessar a listagem de cursos, que se encontra em

<https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/cursos/cursos.html>, e verificar se o curso está listado. Caso sim, deve-se selecionar o curso. Exibido o curso, indicará que o teste fora um sucesso.

### 6.1.5 – Realizar Formulário

**Figura 14-** Crud universidade



Fonte: Próprio autor

Acessando a *URL* [*https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/formulario/formulario.html*](https://icei-puc-minas-pples-ti.github.io/plf-es-2022-1-ti1-7946100-01-g5-entedimento-das-areas-de-ti/codigo/views/formulario/formulario.html), deve-se deparar com a tela de formulários. Esta tela é responsável pelo teste vocacional, coletando dados do usuário por meio de perguntas, definirá qual dos cursos cadastrados melhor se enquadra.

Para o teste, será necessário o preenchimento do formulário múltiplas vezes, com o intuito de identificar se o sistema está indicando cursos diferentes a cada resposta. Caso positivo, o teste fora um sucesso.

# Referências bibliográficas

1. TRELLO. Trello. Disponível em: [www.trello.com](http://www.trello.com/). Acesso: 04 jul. 2022
2. GitHub. GitHub. Disponível em: [www.github.com](http://www.github.com/). Acesso: 04 jul. 2022
3. Roger-that flow: um uso aprimorado do git-flow. Daniel Almeida Luz. Disponível em: <https://www.bdm.unb.br/bitstream/10483/21938/1/2018_DanielAlmeidaLuz_tcc.pdf> . Acesso: 04 jul. 2022
4. BOOTSTRAP. About. Disponível em: <https://getbootstrap.com/docs/5/about/overview/>. Acesso em: 04 jul. 2022.
5. Jquery. Jquery. Disponível em: <https://jquery.com/> . Acesso em: 04 jul. 2022
6. DATATABLES. DataTables. Disponível em: <https://datatables.net/>. Acesso em: 04 jul. 2022.
7. OFICINAS DE ALGORITIMOS COM ESTUDANTES DO SEGUNDO GRAU: UMA PESQUISA AÇÃO. Carlos Eduardo, Dagoberto Melo. Disponível em: <https://repositorio.animaeducacao.com.br/bitstream/ANIMA/13748/1/TCC%20-%20Carlos%20e%20Dagoberto.pdf> . Acesso em: 04 jul. 2022.

1. Metodologia ágil é uma forma de acelerar as entregas durante o desenvolvimento de um projeto, fracionando o todo em entregas incrementais. [↑](#footnote-ref-1)