

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

EDUARDO JEAN FALARZ

**CAMPUS VIRTUAL V2: REPENSANDO A UTFPR SIMULADA DE
GUARAPUAVA**

GUARAPUAVA

2026

EDUARDO JEAN FALARZ

**CAMPUS VIRTUAL V2: REPENSANDO A UTFPR SIMULADA DE
GUARAPUAVA**

**VIRTUAL CAMPUS V2: RETHINKING THE VIRTUAL UTFPR OF
GUARAPUAVA**

Projeto de Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Andres Jessé Porfirio

GUARAPUAVA

2026



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

RESUMO

O presente trabalho apresenta a reestruturação e modernização do Campus Virtual da UTFPR de Guarapuava, uma plataforma digital voltada para a divulgação institucional e exploração imersiva da infraestrutura universitária. O problema central reside nas limitações de flexibilidade e no alto acoplamento de dados da versão anterior, o que dificultava a atualização de conteúdos por usuários sem conhecimentos técnicos em programação. O objetivo principal consiste no desenvolvimento de uma nova plataforma composta por uma interface 3D interativa baseada em WebGL e um painel administrativo para a gestão dinâmica e centralizada de dados. A metodologia fundamenta-se na coleta de requisitos junto aos setores de comunicação e tecnologia da instituição, seguida pela priorização de funcionalidades por meio do método MoSCoW. O desenvolvimento tecnológico utiliza o framework React para o front-end, a biblioteca Three.js para a renderização tridimensional e o PocketBase para a persistência de dados. O principal resultado esperado é a entrega de um Produto Mínimo Viável (MVP) que confira autonomia aos editores para gerenciar menus, *links* e elementos da maquete virtual de forma simplificada. Conclui-se que a principal contribuição deste trabalho é a criação de uma infraestrutura genérica e flexível, capaz de ser adaptada por outros campus universitários, fortalecendo a comunicação estratégica entre a instituição e a sociedade por meio de ferramentas lúdicas e interativas.

Palavras-chave: campus virtual; gestão de informações; renderização web; divulgação institucional; realidade virtual.

ABSTRACT

This work presents the restructuring and modernization of the UTFPR Guarapuava Virtual Campus, a digital platform aimed at institutional visibility and immersive exploration of university infrastructure. The central problem lies in the flexibility limitations and high data coupling of the previous version, which hindered content updates by users without technical programming knowledge. The main objective consists of developing a new platform composed of a 3D interactive interface based on WebGL and an administrative panel for dynamic and centralized data management. The methodology is based on requirements gathering with the institution's communication and technology sectors, followed by functional prioritization through the MoS-CoW method. The technological development utilizes the React framework for the front-end, the Three.js library for three-dimensional rendering, and PocketBase for data persistence. The main expected result is the delivery of a Minimum Viable Product (MVP) that provides autonomy for editors to manage menus, links, and virtual model elements in a simplified manner. It is concluded that the main contribution of this work is the creation of a generic and flexible infrastructure, capable of being adapted by other university campuses, strengthening strategic communication between the institution and society through playful and interactive tools.

Keywords: virtual campus; information management; web rendering; institutional visibility; virtual reality.

LISTA DE FIGURAS

Figura 1 – Ilustração da arquitetura do sistema	16
Figura 2 – Mapa Virtual	22
Figura 3 – Mapa Virtual com Menu Exposto	23
Figura 4 – Lista de Páginas	24
Figura 5 – Editar Página	25
Figura 6 – Editar Página - Bloco RTF	25
Figura 7 – Editar Página - Bloco de Imagens/Vídeos	26
Figura 8 – Lista de Itens do Menu	26
Figura 9 – Editar Itens do Menu	27
Figura 10 – Editar Itens do Menu - Criar Categoria	28
Figura 11 – Lista das Categorias do Menu	28
Figura 12 – Lista de Entidades - Visão em Lista comum	29
Figura 13 – Nova Entidade - Entidade Individual	30
Figura 14 – Lista de Arquivos 3D (Malhas)	30
Figura 15 – Fragmento dos Usuários	32
Figura 16 – Fragmento do Menu	33
Figura 17 – Fragmento das Entidades	34
Figura 18 – Fragmento dos Elementos/Conteúdos	35
Figura 19 – Exemplo de um diagrama de linha de tempo no sistema antigo (RF 17) .	42
Figura 20 – Exemplo de um cronograma no sistema antigo (RF 18)	42
Figura 21 – Diagrama completo do banco de dados	44

LISTA DE TABELAS

Tabela 1 – Requisitos do mapa interativo	20
Tabela 2 – Requisitos do painel administrativo	21
Tabela 3 – Requisitos globais	21
Tabela 4 – Cronograma de tarefas divididas por meses de 2026	38

LISTA DE ABREVIATURAS E SIGLAS

Siglas

API	<i>Application Programming Interface</i>
ASCOM	Assessoria de Comunicação
CGI	Comitê Gestor da Internet no Brasil
CI/CD	<i>Continuous Integration and Continuous Development</i>
COGETI	Coordenadoria de Gestão de Tecnologia da Informação
DBML	<i>Database Markup Language</i>
JSON	<i>JavaScript Object Notation</i>
MVP	<i>Minimum Viable Product</i>
PaaS	<i>Platform as a Service</i>
PDI	Plano de Desenvolvimento Institucional
RF	Requisitos Funcionais
RNF	Requisitos Não Funcionais
RTF	<i>Rich Text Format</i>
SEI	Seminário de Extensão e Inovação
SGBDR	Sistema de Gerenciamento de Banco de Dados Relacional
SICITE	Seminário de Iniciação Científica e Tecnológica da UTFPR
SQL	<i>Structured Query Language</i>
UI	User Interface
WebGL	<i>Web Graphics Library</i>

SUMÁRIO

1	INTRODUÇÃO	8
1.1	Considerações iniciais	8
1.2	Objetivos	9
1.2.1	Objetivos gerais	9
1.2.2	Objetivos específicos	10
1.3	Justificativa	10
2	O CAMPUS VIRTUAL V2	12
2.1	Escopo geral	12
2.1.1	O mapa interativo	12
2.1.2	O painel administrativo	12
2.2	Definição dos usuários	13
2.2.1	Administradores	13
2.2.2	Editores	13
2.2.3	Visitantes	13
3	METODOLOGIA DO DESENVOLVIMENTO	14
3.1	Coleta e classificação dos requisitos	14
3.2	Arquitetura do sistema	15
3.3	Modelagem de dados e interfaces	16
3.4	Fluxo de desenvolvimento, integração e implantação	17
3.5	Implementação do backend e do painel administrativo	18
3.6	Modelagem 3D e construção do mapa interativo	18
4	RESULTADOS PRELIMINARES	19
4.1	Levantamento e priorização dos Requisitos	19
4.2	Prototipação das telas	22
4.2.1	O mapa virtual	22
4.2.2	O painel administrativo	23
4.3	Prototipação do banco de dados	31
4.3.1	Usuários	31
4.3.2	Menu	32
4.3.3	Entidades	33

4.3.4	Elementos/Conteúdos	34
5	CRONOGRAMA DE TRABALHO	36
6	CONSIDERAÇÕES FINAIS	39
	REFERÊNCIAS	40
	APÊNDICE A EXEMPLOS DA PRIMEIRA VERSÃO DO SISTEMA	42
	APÊNDICE B DIAGRAMA COMPLETO DA MODELAGEM DO BANCO DE DADOS	44

1 INTRODUÇÃO

Este capítulo abriga uma apresentação do contexto geral do projeto e quais os objetivos propostos para resolver o problema apresentado.

1.1 Considerações iniciais

A UTFPR de Guarapuava realiza vestibulares próprios regularmente de forma a selecionar alunos para seus cursos. A divulgação desses processos seletivos se dá de várias maneiras, incluindo a distribuição de panfletos, visitas a escolas e até mesmo ações nas adjacências do campus que contam com a presença de servidores, docentes e discentes. É evidente que cada medida tem sua eficácia; contudo, torna-se inegável que, na atualidade, a internet constitui um dos principais meios de comunicação e divulgação de informações.

Segundo o Comitê Gestor da Internet no Brasil (CGI), 83% dos domicílios brasileiros têm acesso à internet. Além disso, 34% dos usuários de internet entre 16 e 24 anos buscaram informações sobre cursos de graduação, pós-graduação ou de extensão (CGI.br, 2025). Seguindo essa tendência e essas mudanças geracionais, as instituições universitárias enfrentam o desafio de não apenas incrementar sua presença digital, mas fazê-lo de forma a alavancar a retenção da atenção do público geral, em especial dos mais jovens.

A aplicação da estratégia de buscar por canais de comunicação mais atraentes e eficientes entra em consonância com o Plano de Desenvolvimento Institucional (PDI) da UTFPR, que explicita a importância de a comunicação entre a instituição e a sociedade ser tratada como um processo estratégico. Tal abordagem visa impulsionar a divulgação da universidade e de suas ações, além de fortalecer sua imagem e valores junto à comunidade ao disponibilizar informações que permitam a obtenção de dados e a efetiva interação entre o público e a instituição (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ, 2023, p. 302-303).

Como resposta a essa diretriz estratégica, ações focadas no uso de plataformas digitais, sites e demais mídias no ambiente virtual tendem a exercer um grande impacto na população, de modo que a utilização de maquetes virtuais 3D surge como uma alternativa para expandir a capacidade institucional de permear esse campo. Esse tipo de recurso pode auxiliar na transposição de barreiras físicas do campus, permitindo que candidatos, estudantes, visitantes e demais interessados explorem e conheçam a infraestrutura da instituição por meio de uma interface interativa. Além disso, essa abordagem é capaz de aprofundar a imersão do usuário quando comparada aos formatos tradicionais de interação; em vez de textos e imagens estáticas, o público pode observar o ambiente por uma perspectiva diferenciada, interagindo com a maquete da maneira que considerar mais conveniente.

Em 2021, as edições do Seminário de Extensão e Inovação (SEI) e do Seminário de Iniciação Científica e Tecnológica da UTFPR (SICITE)¹ ocorreram no campus de Guarapuava,

¹ <https://seisicite2021.gp.utfpr.edu.br/>

ocasião na qual nasceu a concepção de uma maquete virtual. A primeira versão do campus virtual tinha por objetivo centralizar todas as informações referentes aos eventos em um único local, contando com menus estruturados para *links*, cronogramas e detalhes sobre os trabalhos desenvolvidos e expostos. Contudo, sua implementação dispunha também de outro elemento que a tornou singular e de grande apelo visual: uma representação interativa do ambiente físico da instituição. Nessa plataforma, era possível não apenas navegar pelos blocos e conhecer a infraestrutura da sede anfitriã, mas também participar de um tour virtual pela capital nacional da cevada e do malte.

Cumprido seu papel, a primeira versão do projeto foi lançada abertamente sob a licença MIT (Open Source Initiative, 2026) e posteriormente adaptada para uso como ferramenta de divulgação do vestibular e de outras informações institucionais — o que propiciou o surgimento de algumas ideias para aperfeiçoamento. A Assessoria de Comunicação (ASCOM), responsável pelo diálogo com a comunidade externa, demonstrou interesse no sistema e propôs aprimoramentos em alguns de seus aspectos. A funcionalidade mais solicitada foi uma interface capaz de possibilitar a alteração e a criação dos conteúdos dispostos na maquete 3D, tais como textos, vídeos e fotografias. A primeira versão demanda que todas as edições no material exposto sejam realizadas por meio de solicitação à Coordenadoria de Gestão de Tecnologia da Informação (COGETI), que relatou dificuldades em executar essa e outras atividades pertinentes à manutenção do sistema em razão do alto acoplamento entre dados e interface.

Diante do exposto, o presente trabalho propõe uma reformulação do campus virtual, efetivamente construindo uma segunda versão. Desta forma, busca-se solucionar os principais problemas notados pelos utilizadores da primeira versão do sistema. Para além disso, objetiva-se criar uma estrutura adaptável e reutilizável por quaisquer outras entidades para criação de maquetes virtuais que possam ou não ter relação com o campus de Guarapuava.

1.2 Objetivos

As metas pretendidas para este projeto serão apresentadas nesta seção.

1.2.1 Objetivos gerais

Desenvolver uma plataforma para o Campus Virtual da UTFPR, composta por uma interface imersiva em 3D e um backend para a gestão dinâmica de dados, permitindo a edição centralizada de conteúdos através de um painel administrativo.

1.2.2 Objetivos específicos

- Levantar e classificar os requisitos funcionais e não funcionais do sistema junto à AS-COM e à COGETI, delimitando o escopo do *Minimum Viable Product* (MVP);
- Projetar a arquitetura estrutural do sistema de forma a garantir o desacoplamento entre seus elementos principais;
- Modelar o esquema lógico de dados e prototipar as interfaces de usuário de alta fidelidade para a validação prévia dos fluxos de navegação;
- Estabelecer o fluxo de desenvolvimento e a automação das esteiras de CI/CD, delineando padrões de código, configurando ferramentas para análise estática, testes automatizados e implantação (*deploy*);
- Implementar a camada de persistência de dados no backend e desenvolver o painel administrativo modular baseado em blocos para a gestão centralizada de autenticação, mídias e conteúdos dinâmicos do sistema;
- Modelar os elementos gráficos tridimensionais do campus e construir a interface do mapa interativo para a web.

1.3 Justificativa

Embora seu legado seja importante, existem lacunas não preenchidas e requisitos que hoje são necessários, mas que não estavam presentes na sua concepção.

O principal problema a ser solucionado é o alto acoplamento entre conteúdo dinâmico (textos, vídeos, *links*, menus, entre outros) e interface. Cada um destes elementos do campus virtual é inserido de forma estática, exigindo a alteração dos dados diretamente no código-fonte e a subsequente reimplatação da aplicação no servidor. Isto torna indispensável a atuação de pessoas com, no mínimo, noções de programação e dificulta a sua manutenção.

Muito próximo ao problema da gestão do conteúdo dinâmico, também é possível observar limitações quanto a transformação de como a maquete 3D do campus virtual é construída e quais interações estão disponíveis aos usuários. Dentro deste tópico, existem duas problemáticas centrais. A primeira é a utilização de um arquivo de modelo 3D único que engloba integralmente o mapa. Desta forma, qualquer alteração, por menor que seja, passa necessariamente pela alteração deste único arquivo e a atualização da maquete inteira. O segundo ponto é mais uma vez o acoplamento entre dados e interface. A primeira versão do campus virtual utiliza um tamanho fixo para a maquete dentro do navegador e estipula pontos fixos onde interações podem ocorrer. Um exemplo são as janelas com informações sobre determinados blocos que são abertas quando há um clique na entidade que representa aquele bloco em questão. Den-

tro do código, isso se traduz como um elemento interagível com altura, largura e coordenadas pré-estabelecidas.

Além disso, a primeira versão do campus virtual carece de uma estrutura automatizada para verificação de integridade (testes e *linters*) e implantação (*deploy*) automatizados, e isso coloca ainda mais empecilhos à sua manutenção.

2 O CAMPUS VIRTUAL V2

O Campus Virtual V2 surge como uma ferramenta versátil de divulgação para a UTFPR, superando as limitações operacionais da versão anterior. O núcleo da solução reside em um painel administrativo centralizado para gestão de dados, eliminando o alto acoplamento entre o conteúdo dinâmico e o mapa virtual, bem como a consequente complexidade de manutenção do código-fonte.

Nessa arquitetura, a maquete virtual atua estritamente como motor de renderização, consumindo dados assincronamente via *Application Programming Interface* (API). Essa separação confere flexibilidade para implementar novas formas de visualização e interatividade no mapa. Ademais, o projeto conta com a possibilidade de reutilização em outros contextos operacionais e com diferentes malhas 3D, extrapolando o campus de Guarapuava. Por fim, essa estrutura modular estabelece a base tecnológica para que a plataforma atue, futuramente, como um motor para edição de cenas tridimensionais¹. Essa evolução permitirá a manipulação direta dos objetos na interface gráfica do painel administrativo, mitigando a dependência do cadastro manual de coordenadas cartesianas (X , Y e Z).

2.1 Escopo geral

O projeto em si é composto por duas partes fundamentais: o mapa interativo da instituição e o painel para administração interna.

2.1.1 O mapa interativo

O mapa possibilita a navegação por uma maquete virtual 3D das dependências da UTFPR de Guarapuava e uma interface de menus com *links* ou informações sobre temas diversos (como eventos e vestibulares). Além disso, a maquete possui interações a partir do clique/toque na tela, revelando conteúdos vinculados aos elementos presentes na tela (ex.: blocos do campus).

2.1.2 O painel administrativo

Este recurso nada mais é que uma interface onde os conteúdos contidos na maquete virtual são gerenciados. Os menus e sua organização, os *links* e demais informações (vídeos,

¹ Vale pontuar que a intenção não consiste em recriar softwares de modelagem baseados no manejo de polígonos, mas sim facilitar o arranjo espacial de uma cena 3D previamente desenvolvida, permitindo a inclusão, exclusão e alteração de propriedades de translação, rotação e escala das entidades diretamente na cena gráfica, com a consequente atualização automática de suas coordenadas cartesianas.

imagens, textos, diagramas, etc) são criados e modificados a partir daqui. Ademais, usuários especiais - que serão detalhados na próxima seção - são capazes de administrar os elementos presentes na maquete em si (ex.: blocos do campus), modificando os valores de transformação espacial (translação, rotação e escala) de cada um.

2.2 Definição dos usuários

Três categorias de usuários foram definidas. São eles: administradores, editores e visitantes.

2.2.1 Administradores

Responsáveis por todo o gerenciamento do sistema a partir do painel administrativo. Possuem amplos poderes e sua função principal é a de transformar a maquete virtual ao organizar seus arquivos e metadados, além de manusear os dados referentes às demais informações da maquete, como imagens, vídeos, textos e diagramas.

2.2.2 Editores

São capazes de criar e editar conteúdo (textos, *links*, menus, imagens, etc.) a partir do painel administrativo. Ao contrário dos administradores, este grupo não possui acesso aos mecanismos de transformação espacial (relacionado ao posicionamento de entidades e no desenho do mapa 3D).

2.2.3 Visitantes

Não possuem qualquer acesso ao painel administrativo. Seu campo de ação é exclusivamente focado na maquete virtual. Visitantes podem desfrutar da navegação e das interações disponibilizadas pelo maquete da forma como lhes aprouver e sem qualquer tipo de autenticação exigida.

3 METODOLOGIA DO DESENVOLVIMENTO

Este capítulo é composto por uma apresentação de todos os métodos empreendidos para a construção da aplicação e os materiais usados ao longo do percurso. Ressalta-se que a revisão gramatical e o refinamento estilístico do texto contaram com o auxílio de ferramentas de Inteligência Artificial da Google, em específico o Gemini (GOOGLE, 2026), conforme as orientações da Portaria CNPq nº 2.664/2026 (BRASIL, 2026) e as recomendações institucionais da UTFPR Guarapuava, mantendo-se a responsabilidade integral do autor sobre o conteúdo técnico apresentado.

3.1 Coleta e classificação dos requisitos

A coleta de requisitos fundamenta-se em duas fontes principais. A primeira consiste no sistema precursor e no diálogo com o docente responsável por sua concepção. O objetivo desta etapa é estabelecer os requisitos preliminares através da análise técnica da estrutura legada e de consultas ao administrador original.

A segunda fonte constitui-se por órgãos do campus Guarapuava — a saber: ASCOM e COGETI. A consulta à ASCOM justifica-se por sua competência na divulgação institucional e experiência prévia com a ferramenta anterior. Considera-se que tal experiência tem o potencial de proporcionar críticas construtivas e a proposição de novas funcionalidades. Adicionalmente, inclui-se a COGETI no processo, dada sua proximidade com a infraestrutura tecnológica do campus, visando viabilizar sugestões de caráter técnico para o projeto.

A classificação dos requisitos prevê o uso do método MoSCoW (BERAM, 2023). Esta técnica classificatória consiste em quatro camadas, descritas a seguir:

- **Must have:** são os requisitos essenciais do sistema - aqueles cuja presença é de suma importância para o MVP;
- **Should have:** elementos que devem estar presentes e são importantes para agregar valor ao produto final, mas que não são estritamente necessários nas primeiras versões;
- **Could have:** esta camada, por sua vez, é composta por todos aqueles requisitos que agregam valor, mas que são completamente dispensáveis frente às camadas superiores e podem ser ignorados até que os pontos mais importantes estejam consolidados;
- **Won't have:** Esta categoria ostenta as ideias cujo esforço não reflete igualmente em agregação de valor ao produto final. São requisitos cujo desenvolvimento é descartado na primeira versão do sistema, mas com possibilidade de reavaliação futura.

São considerados três principais pilares para definir a ordem de importância dentro do MoSCoW:

- Requisitos cuja ausência implicaria necessariamente em um funcionamento parcial do projeto (ex.: mapa navegável do campus);
- Funcionalidades já existentes na primeira versão do projeto;
- Percepções sobre as deficiências e espaços de melhoria da primeira versão do projeto sob a ótica da ASCOM e da COGETI.

3.2 Arquitetura do sistema

É possível destacar duas arquiteturas diferentes dentro do projeto em níveis distintos. A primeira diz respeito ao projeto como um todo e ela surge a partir da escolha dos dois elementos principais: o *frontend* em React¹ e o *backend* com PocketBase². Esse conjunto é muito poderoso pois o React é uma biblioteca que potencializa a reutilização de elementos e a dinamização de interfaces. Aliado a ele, o PocketBase é um conjunto de ferramentas como autenticação, gerenciamento de dados e uma API REST para sua utilização por serviços externos. Essa configuração demanda uma arquitetura cliente-servidor a partir da API exposta nativamente pela ferramenta do *backend*.

A segunda diz respeito à comunicação dentro do próprio *frontend* entre a biblioteca de User Interface (UI) React e o *Web Graphics Library* (WebGL)³, que consiste em uma API JavaScript que possibilita a renderização de modelos 3D diretamente no navegador, dispensando downloads adicionais ou plugins. Além disso, essa ferramenta possui uma consolidada integração com o React, viabilizando a renderização dos modelos para o usuário em sincronia com a interface. Todavia, comunicar diretamente estes dois elementos agrega uma série de problemas, como a alta taxa de renderizações desnecessárias no React devido à natureza de funcionamento do WebGL (baseada em um laço contínuo de atualização de quadros e processamento de matrizes na GPU). Ademais, o acoplamento também incorre em uma alta dependência entre as duas partes do *frontend* pela necessidade do uso de nomeações diretas às referências dentro da árvore de entidades da maquete 3D e informações sobre a cena em si, indo de encontro a uma das premissas centrais do projeto em termos de manutenção. Para sanar estes distúrbios, surge a necessidade de uma arquitetura orientada a eventos a partir de um *event bus*. Aliado a isso, integra-se também a biblioteca Three.js⁴ para abstrair a natureza complexa do WebGL (baseada no gerenciamento e cálculo manual de matrizes) — alavancando a produtividade e facilitando a manutenção futura do projeto.

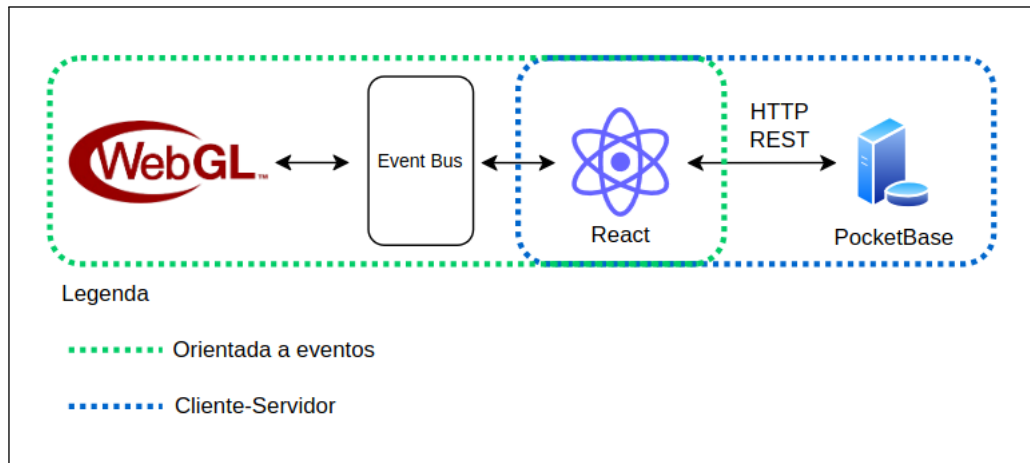
¹ <https://www.epicreact.dev/what-is-react>

² <https://pocketbase.io/docs/>

³ <https://www.khronos.org/webgl/>

⁴ https://developer.mozilla.org/pt-BR/docs/Glossary/Three_js

Figura 1 – Ilustração da arquitetura do sistema



Fonte: Autoria própria, 2026.

Na figura acima, é possível observar uma representação das duas arquiteturas mencionadas. A biblioteca Three.js, apesar de mencionada, foi abstraída.

Por fim, mas não menos importante, adota-se uma estrutura monorepositório com Docker⁵ para o empacotamento da aplicação. Para o auxílio na orquestração dos contêineres, usa-se o Coolify⁶ — um *Platform as a Service* (PaaS) de código aberto, viabilizando uma estrutura autohospedada sem custos operacionais de licenciamento.

3.3 Modelagem de dados e interfaces

O design de interface baseia-se na criação de protótipos de alta fidelidade por meio do Figma⁷, ferramenta amplamente utilizada no mercado para criação de protótipos e designs de aplicações no geral. O procedimento consiste na elaboração visual das telas utilizando como base os elementos da biblioteca Mantine⁸, que é um conjunto de componentes padronizados que facilitam a construção de interfaces. Essa etapa busca validar o fluxo de navegação antes do início do desenvolvimento.

A modelagem do banco de dados fundamenta-se no uso da ferramenta DBDiagram⁹, que segue os padrões e a sintaxe da *Structured Query Language* (SQL). A escolha desta ferramenta é vinculada à sua abordagem de uso que dispensa instalações por ser acessível pelo navegador e a facilidade de criação dos diagramas lógicos a partir da linguagem de marcação *Database Markup Language* (DBML). O procedimento compreende a definição formal das coleções (tabelas), atributos e seus respectivos relacionamentos, resultando em um esquema lógico que orienta a implementação da camada de persistência de dados.

⁵ <https://docs.docker.com/get-started/docker-overview/>

⁶ <https://absam.io/blog/tudo-sobre-coolify/>

⁷ <https://help.figma.com/hc/en-us/articles/14563969806359-What-is-Figma>

⁸ <https://mantine.dev/about/>

⁹ <https://docs.dbdiagram.io/>

3.4 Fluxo de desenvolvimento, integração e implantação

O desenvolvimento do projeto é a etapa que consolida o planejamento delineado até então. Dessa forma, é preciso definir certas ferramentas e métodos para que o processo seja íntegro e uniforme.

Para administrar e armazenar todo o código desenvolvido durante o percurso deste projeto, empregam-se as ferramentas Git¹⁰ e GitHub¹¹ por sua confiabilidade, facilidade de uso e ampla adoção no mercado de software. Sob a égide do GitHub, são destacáveis ainda o GitHub Actions¹² — para automações referentes a testes e verificações de padronização de código — e o GitHub Projects¹³ — para a estruturação e organização das tarefas dentro do projeto.

O método de versionamento empregado ao longo do desenvolvimento é o *gitflow* (ATLASSIAN, 2026). Este método é adotado para garantir a integridade do código-fonte e a rastreabilidade das implementações realizadas. Sob este padrão, cada requisito técnico origina uma ramificação (*branch*) específica, adotando a convenção de nomenclatura:

```
@git_user/escopo/tipo_de_ramificação/numero_tarefa-nome_-tarefa.
```

Após a conclusão de uma funcionalidade, submete-se uma *pull request* para revisão do docente orientador. Neste momento, uma esteira automatizada para checagem de integridade atua para ajudar a evitar regressões — isto é, a quebra de funcionalidades preexistentes. Essa ação executa ferramentas específicas para checagem estática da qualidade do código com ESLint¹⁴ e testes com Jest¹⁵, *React Testing Library*¹⁶ e Playwright¹⁷.

O fluxo de desenvolvimento encerra-se com a integração do código à *branch* de desenvolvimento (*development*) ou com o retorno da tarefa para ajustes, conforme as observações da revisão técnica.

A partir disso, entram em ação as automações de implantação. Neste cenário, o emprego do Docker para empacotamento e do Coolify para o gerenciamento da implantação e da aplicação tornam erros menos prováveis e reforçam a padronização de processos.

¹⁰ <https://git-scm.com/about>

¹¹ <https://github.com/about>

¹² <https://github.com/features/actions>

¹³ <https://docs.github.com/pt/issues/planning-and-tracking-with-projects/learning-about-projects/about-projects>

¹⁴ <https://www.alura.com.br/conteudo/eslint-padronizando-codigo-regras-personalizadas>

¹⁵ <https://www.devmedia.com.br/teste-unitario-com-jest/41234>

¹⁶ <https://www.freecodecamp.org/portuguese/news/react-testing-library-tutorial-com-exemplos-de-codigo-em-javascript/>

¹⁷ <https://www.horadoqa.com.br/articles/playwright.html>

3.5 Implementação do backend e do painel administrativo

Definido o PocketBase como o *backend*, implementa-se o diagrama lógico criado na etapa 3.3 (Modelagem de dados e interfaces). O esquema elaborado previamente integra-se perfeitamente com a estrutura da ferramenta, uma vez que o sistema utiliza internamente o Sistema de Gerenciamento de Banco de Dados Relacional (SGBDR) SQLite. Contudo, a plataforma gerencia o esquema de dados por meio de uma interface abstrata baseada em coleções, assemelhando-se operacionalmente a ferramentas NoSQL orientadas a documentos. Por isso, é comum considerar o PocketBase como uma solução híbrida.

Além do gerenciamento de persistência, outro módulo fundamental do *backend* consiste na gestão de regras de acesso à API. Este mecanismo dita quais perfis estão aptos para consumir, criar e alterar certos tipos de registros. Isso garante uma separação clara de papéis e impede o acesso indevido a recursos restritos.

Uma vez estabelecidas as diretrizes de segurança no *backend*, compõem-se os componentes responsáveis pela listagem, criação e modificação das páginas de conteúdo. Adota-se uma abordagem modular, permitindo que cada página abrigue diferentes estruturas informacionais — como *Rich Text Format* (RTF), arquivos de mídia e diagramas cronológicos — dispostas verticalmente e conforme a necessidade. De forma semelhante, executa-se a materialização dos elementos responsáveis pela visualização, elaboração e edição das entidades, itens do menu, suas categorias e arquivos de malha (*meshes*) — estabelecendo os vínculos lógicos necessários para utilização na camada gráfica do sistema.

3.6 Modelagem 3D e construção do mapa interativo

Conectada à infraestrutura de renderização, a alimentação gráfica desse ecossistema ocorre por meio do software Blender¹⁸, um estúdio de código aberto voltado para a modelagem de malhas (*meshes*) 3D. Essa ferramenta adequa-se às necessidades do projeto devido à alta disponibilidade de documentação e à sua capacidade nativa de exportar arquivos no formato glTF/GLB, que se consolidou como o padrão ideal para transmissão e carregamento eficiente de cenas tridimensionais na Web via Three.js. Com todas as partes estruturadas, a maquete virtual é efetivamente montada. As modelagens 3D alimentam sua construção a partir da inserção no painel administrativo, registrando-se também os seus metadados.

¹⁸ <https://www.blender.org/about/>

4 RESULTADOS PRELIMINARES

Esta seção reúne os resultados alcançados até então. De forma a organizar a leitura, os conteúdos foram categorizados em três partes, começando pela categorização de requisitos (seção 4.1), prototipação das interfaces visuais (seção 4.2) e, por fim, pela montagem do esquema do banco de dados (seção 4.3).

4.1 Levantamento e priorização dos Requisitos

A partir do levantamento dos requisitos conforme delineado na seção 3.1 (Coleta e classificação dos requisitos), chegou-se à conclusão de que o sistema antigo não possui as capacidades essenciais para suportar as necessidades atuais de divulgação e comunicação com a comunidade (tanto interna como externa) do campus.

Dentre os apontamentos feitos pelos responsáveis pela manutenção do sistema (como o COGETI ou o próprio professor criador do projeto), estão a falta de maleabilidade do sistema, dificuldade de preservação, expansão e modificação - tanto de conteúdos como páginas e *links* como de artes e animações. Além disso, o sistema carece de mecanismos de *Continuous Integration and Continuous Development* (CI/CD) para automatizar processos tanto de checagem de qualidade e padronização de código como de implantação.

As observações por parte dos encarregados da criação de conteúdos para o campus virtual (como a ASCOM) partem de uma mesma raiz: a inexistência de um painel administrativo que possibilitasse a criação, edição e deleção de conteúdos - como páginas, fotos, vídeos, textos, menus etc.

Além do levantamento dos problemas da primeira versão, o diálogo com as partes interessadas foi muito frutífero no que concerne a novas ideias para o futuro do sistema. Com todas estas informações em mãos, iniciou-se a priorização dos requisitos propriamente dita. Abaixo, serão apresentados estes itens segregados em virtude das categorias abaixo delineadas sob a perspectiva da distinção entre Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF) (Casa do Desenvolvedor, 2023):

- **Requisitos do mapa interativo (Tabela 1):** todos que fazem jus ao mapa interativo e suas funcionalidades.
- **Requisitos do painel administrativo (Tabela 2):** aqueles pertinentes ao painel administrativo, acessado por editores/administradores.
- **Requisitos globais (Tabela 3):** Ideias que não são necessariamente conectadas exclusivamente a um dos itens acima.

Tabela 1 – Requisitos do mapa interativo

ID	Requisito	MoSCoW
RF01	Mapa navegável do campus.	Must
RF02	Navegação restrita na perspectiva de um ponto fixo no mapa.	Must
RF03	Menu navegável (<i>links</i> e abertura de elementos).	Must
RF04	Elementos interativos (ex.: painel com fotos, textos e cronogramas de um determinado bloco que aparecem ao clicar em cima do bloco em questão).	Must
RF05	Design Responsivo: a interface deve se adaptar a diferentes resoluções e dispositivos (Desktop/Mobile).	Must
RF06	Carregamento assíncrono de elementos com tela de progresso.	Must
RF07	Tutorial de uso que ensine a como navegar pelo mapa e utilizar as ferramentas disponíveis.	Should
RF08	Sistema de busca/filtragem de entidades (ex.: buscar bloco B).	Should
RF09	Rotas dinâmicas que possibilitem abrir conteúdo de um elemento ao entrar por um link específico (ex.: site.com/mapa/bloco-b abre o conteúdo atrelado ao bloco B).	Should
RF10	Visão interna 2D dos edifícios (ex.: uma planta baixa).	Could
RF11	Visão 3D (navegação em todos os eixos).	Won't
RF12	Botão para reiniciar a posição da câmera.	Won't
RF13	Visão interna 3D dos edifícios, tal como a parte exterior.	Won't
RF14	Simulação dinâmica de iluminação solar, afetando a sombra dos edifícios e demais elementos.	Won't

Fonte: Autoria própria (2026).

Tabela 2 – Requisitos do painel administrativo

ID	Requisito	MoSCoW
RF15	Suporte à edição de textos, diagramas, vídeos, fotos e tabelas por meio de um painel administrador.	Must
RF16	Suporte a RTF, uma ferramenta que possibilita vários tipos de formatação e estilização de textos de forma integrada.	Must
RF17	Painel administrativo para mudanças e adições (textos, fotos, modelos etc.).	Must
RNF01	Sistema de permissões no painel administrativo (entre funções permitidas a editores e administradores).	Must
RF18	Suporte à criação livre de diagramas de linha de tempo (ver Figura 19).	Should
RF19	Suporte à criação livre de cronogramas (ver Figura 20).	Should
RF20	Modo claro da interface (esquema de cores).	Should
RF21	Substituição e adição de arquivos 3D pelo painel administrativo.	Could
RF22	Criação de pontos de interesse no mapa a partir do painel administrativo de forma dinâmica.	Won't

Fonte: Autoria própria (2026).

Tabela 3 – Requisitos globais

ID	Requisito	MoSCoW
RNF02	Implantação automatizada	Should
RNF03	Cache dos arquivos 3D localmente para otimizar visitas futuras.	Could

Fonte: Autoria própria (2026).

4.2 Prototipação das telas

Ao todo, foram projetadas 15 (quinze) telas distintas, das quais 13 (treze) são específicas do painel administrativo, enquanto duas dizem respeito ao mapa virtual. Embora este número possa parecer desproporcional, o mapa virtual possui um número menor de telas prototipadas pois sua natureza operacional difere muito daquela apresentada no painel administrativo. As duas telas desenvolvidas ilustram o modelo prévio do mapa virtual, suas ferramentas de ação e o menu. No entanto, o escopo deste ambiente comporta um volume praticamente infinito de estados visuais, justificado precisamente pela sua característica interativa.

4.2.1 O mapa virtual

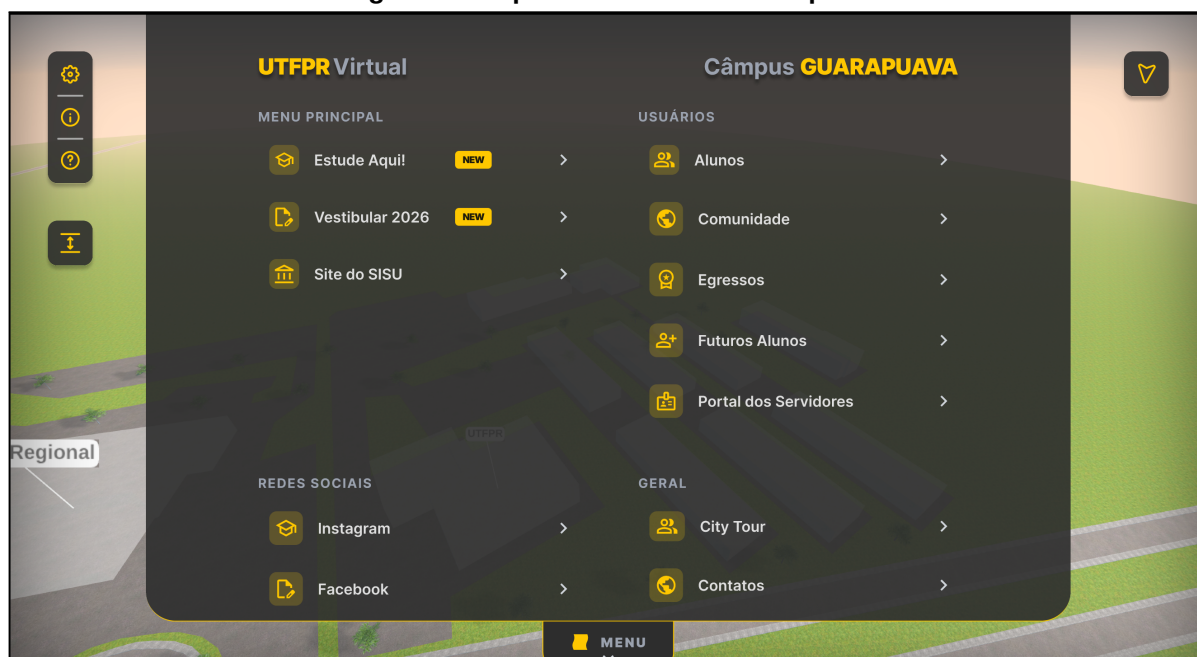
Ao abrir a maquete 3D, o usuário depara-se com o mapa virtual. Dentro desta tela, existem botões de ação (para acessar as configurações, o tutorial e para o controle de câmera). No topo, existe o botão que aciona o menu que contém *links* e outras informações (ver Figura 2 e Figura 3).

Figura 2 – Mapa Virtual



Fonte: Autoria própria, 2026.

Figura 3 – Mapa Virtual com Menu Exposto



Fonte: Autoria própria, 2026.

4.2.2 O painel administrativo

Dentro do painel administrativo, as ações disponíveis serão controladas pelo tipo de acesso que o usuário possui. Como mencionado antes, usuários administradores tem poderes mais amplos que usuários editores. Para fins de simplificação, as imagens do painel administrativo mostradas abaixo levam em consideração um usuário do tipo administrador, pois as mudanças são mínimas para o usuário editor (apenas os acessos do menu lateral mudam).

A primeira tela vista pelo usuário ao logar é a de gerenciamento de Páginas (ver Figura 4), onde é possível ver a lista de páginas existentes e ainda criar novas.

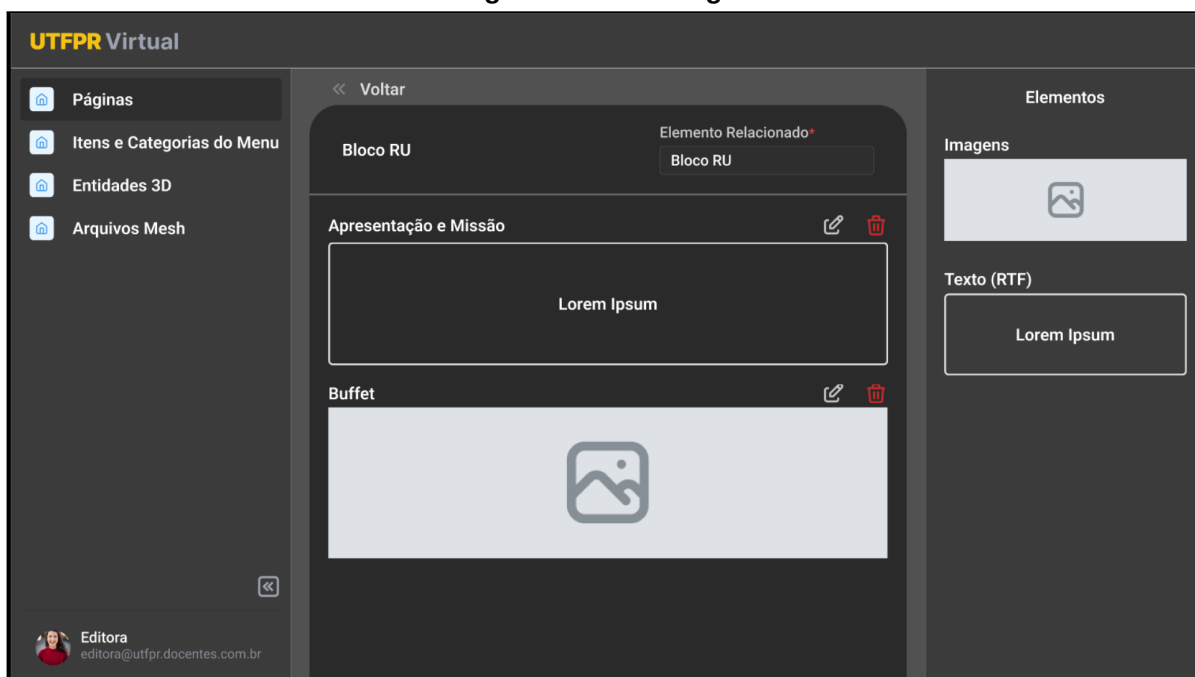
Figura 4 – Lista de Páginas

Nome	Criado em	Modificado em	Tipo relacionado
Bloco RU	01/01/2026	01/01/2026	Elemento 3D
Bloco A	26/01/2026	27/01/2026	Elemento 3D
Nossos Cursos	02/02/2026	02/02/2026	Item de Menu
Portaria	06/02/2026	06/02/2026	Elemento 3D

Fonte: Autoria própria, 2026.

Selecionado o ícone de edição, abre-se o painel para alterações (ver Figura 5). Nesta parte, é possível modificar dados da página - como nome, o elemento ao qual ela está relacionada e os blocos de conteúdo. Os blocos de conteúdo são dispostos conforme a ordem e quantidade definida pelo usuário, podendo ser excluídos, modificados e adicionados de forma livre. No menu lateral direito, é possível ver os elementos de conteúdos disponíveis para utilização.

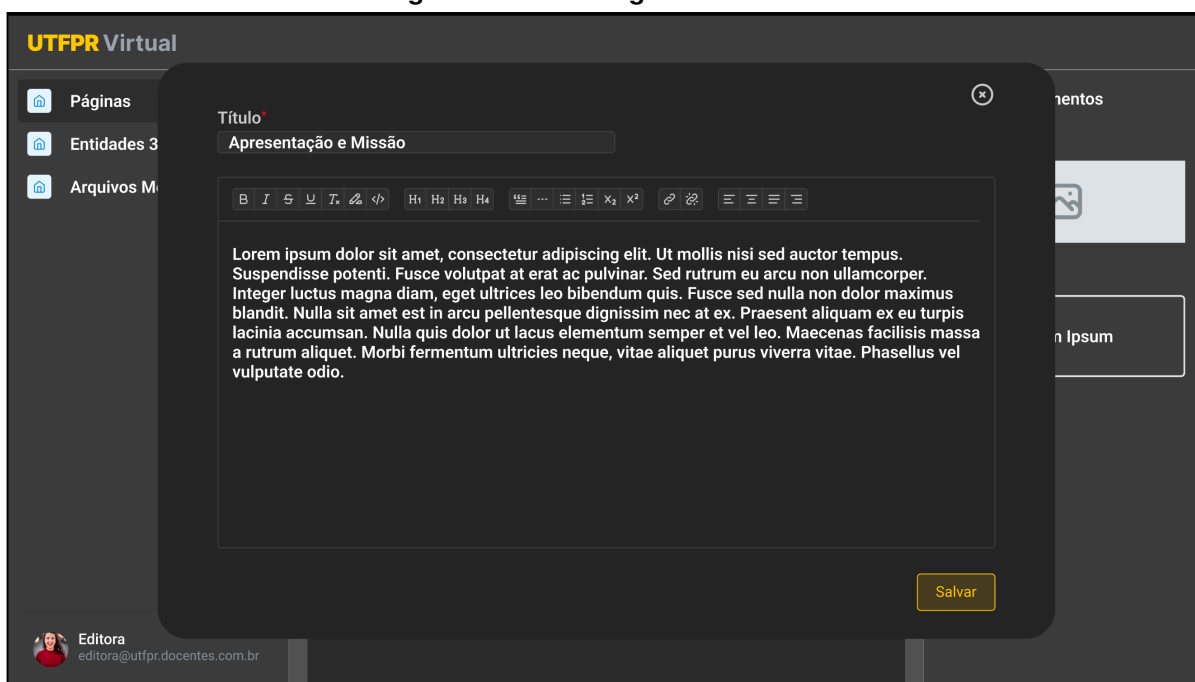
Figura 5 – Editar Página



Fonte: Autoria própria, 2026.

A edição do bloco de texto conta com um editor RTF, que facilita a escrita de textos com diversos recursos diferentes (ver Figura 6).

Figura 6 – Editar Página - Bloco RTF

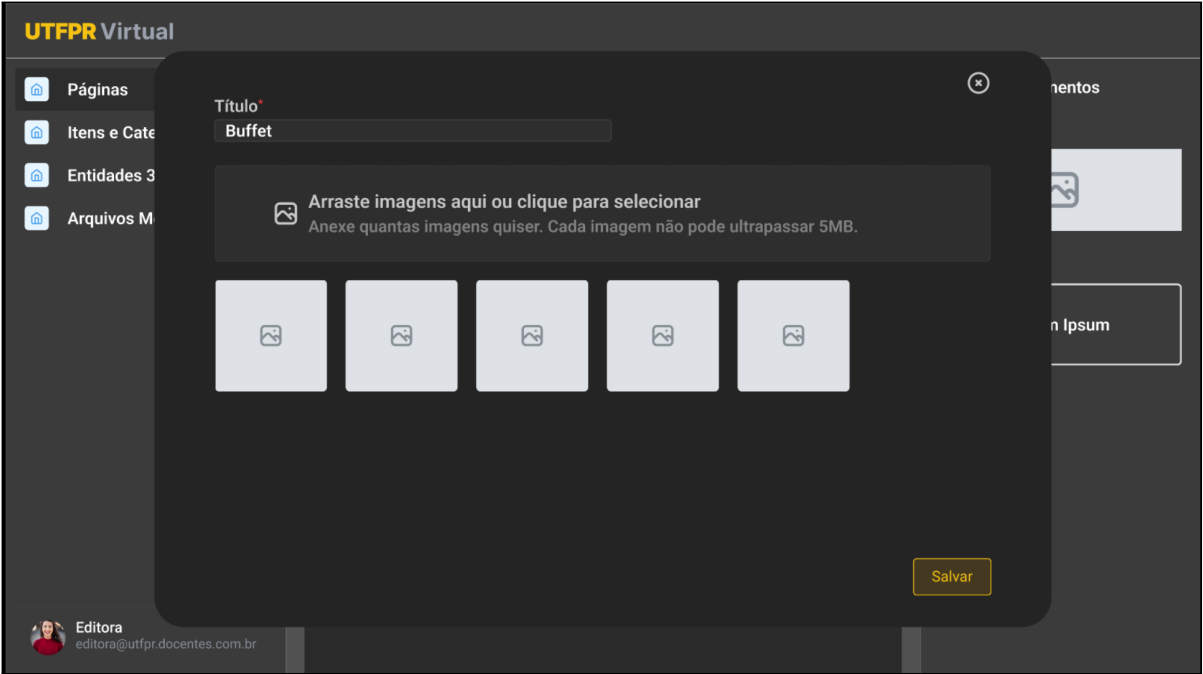


Fonte: Autoria própria, 2026.

A edição do bloco de imagens/vídeos possui um local para o envio dos arquivos (tanto por seleção manual como pelo fluxo "arraste e solte"). Ademais, os arquivos salvos ficam expos-

tos como miniaturas para uma visualização prévia dos conteúdos. Cada um pode ser excluído a qualquer momento com um clique em cima do próprio elemento (ver Figura 7).

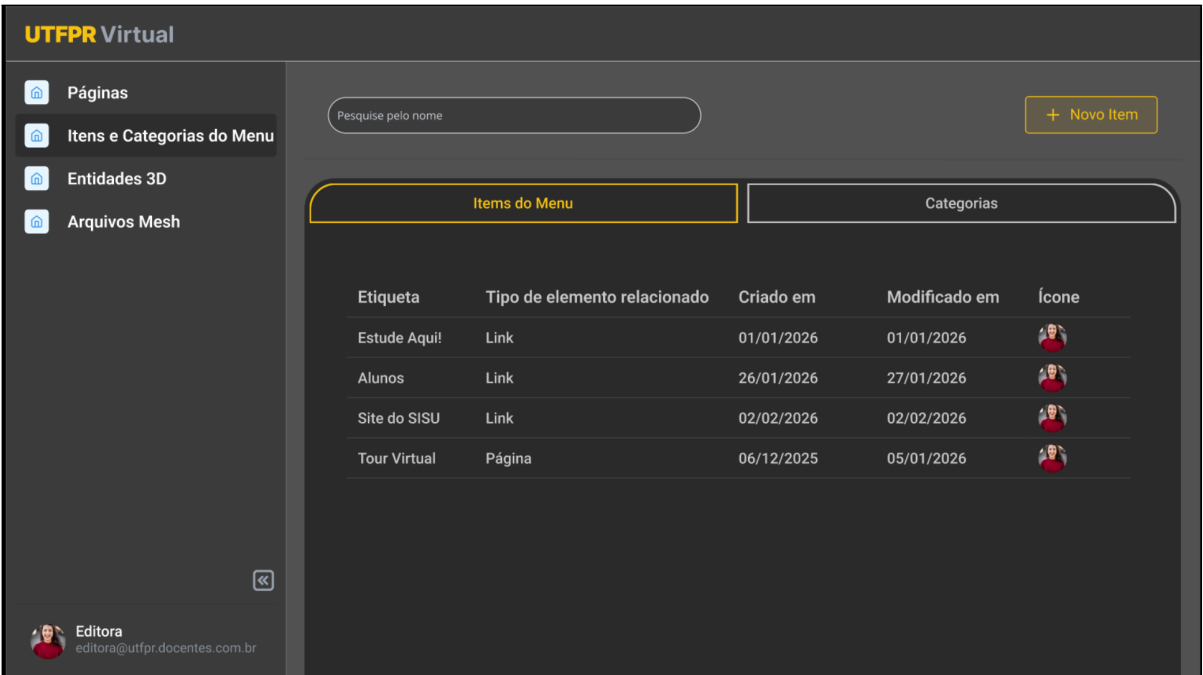
Figura 7 – Editar Página - Bloco de Imagens/Vídeos



Fonte: Autoria própria, 2026.

O segundo módulo possível de ser acessado é o de itens do menu e suas categorias. Aqui, é possível criar e editar itens que serão dispostos no menu do campus virtual (ver Figura 8 e Figura 9), além de organizar estes itens em categorias dinâmicas (ver Figura 11).

Figura 8 – Lista de Itens do Menu



Fonte: Autoria própria, 2026.

Abaixo, a criação/edição dos itens do menu possui algumas configurações importantes. É possível selecionar o tipo de referência do item do menu (bloco de conteúdo ou link). Essa seleção é mutuamente exclusiva, sendo que só um pode ser escolhido para cada item.

É possível ainda aninhar itens dentro de outros itens (neste caso, a categoria é obrigatoriamente a mesma para ambos.)

Além disso, esta tela possui a capacidade de facilitar o fluxo de criação e vinculação de uma categoria ao permitir sua criação por meio de um painel flutuante (ver Figura 10). Também existe uma tela dedicada pra criação de categorias, que não será evidenciada por não possuir elementos de destaque.

Figura 9 – Editar Itens do Menu

UTFPR Virtual

« Voltar

Etiqueta*
example_label

Elemento Vinculado*

Link Página

https://link.example.com Seleccione uma página

É um elemento aninhado?*

Não Sim

Categoria*
Selecione... +

Carregar ícone
Arraste e solte arquivo de imagem do ícone ou clique no botão abaixo para selecioná-lo.

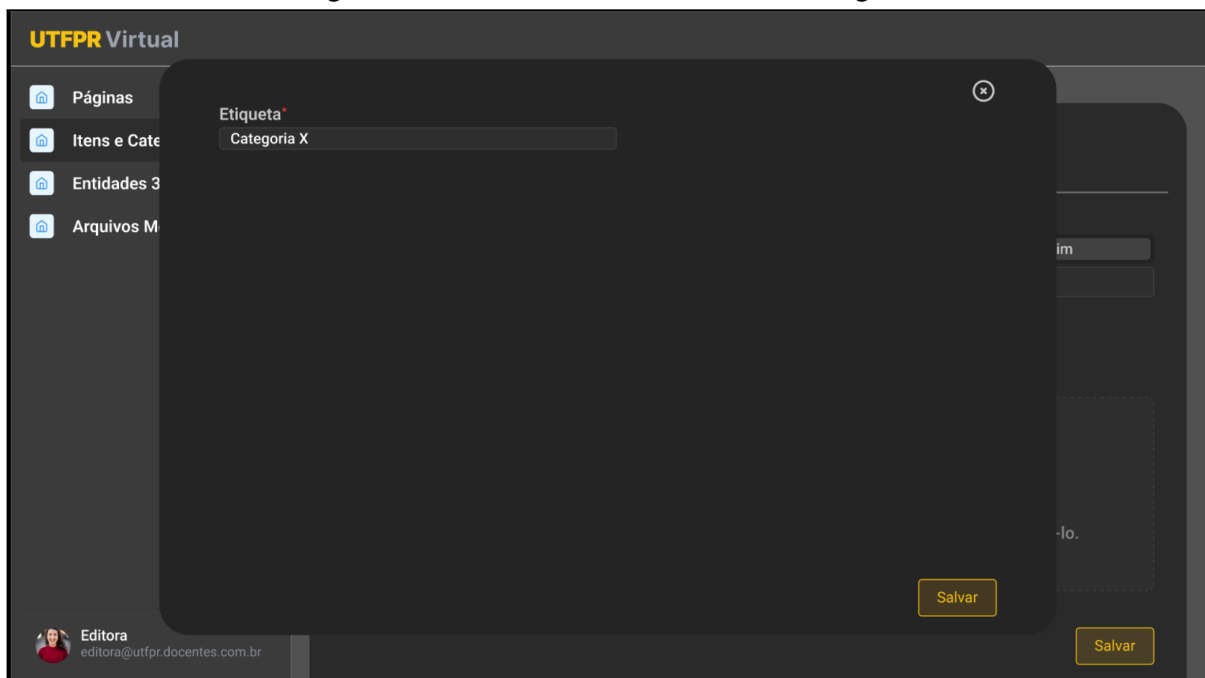
Selecionar Arquivos

Salvar

Editora
editora@utfpr.docentes.com.br

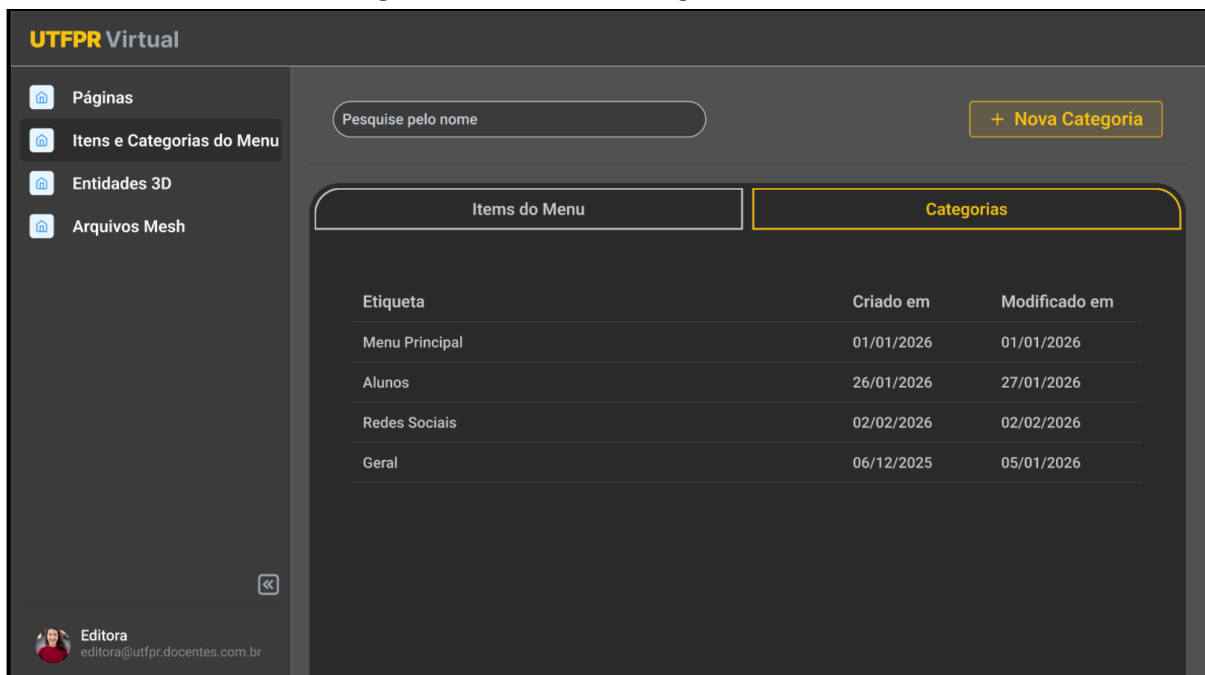
Fonte: Autoria própria, 2026.

Figura 10 – Editar Itens do Menu - Criar Categoria



Fonte: Autoria própria, 2026.

Figura 11 – Lista das Categorias do Menu



Fonte: Autoria própria, 2026.

O terceiro módulo para onde se pode navegar é o de entidades 3D. Nele, é possível interagir com as entidades já existentes ou até criar novas. Existem dois tipos de visualização neste módulo: em uma lista comum e em um formato de edição em um conjunto de *JavaScript Object Notation* (JSON) — ver Figura 12. Nestas imagens, é possível notar um termo específico

empregado na lista: *slug*. Este termo é utilizado neste contexto como um identificador único e legível para humanos que possa representar um elemento específico dentro do campus virtual.

Figura 12 – Lista de Entidades - Visão em Lista comum

Slug	Mesh	Ativo	Criado em	Modificado em
ru_block	Bloco Estudantil Genérico	Sim	01/01/2026	01/01/2026
block_a	Bloco Estudantil Genérico	Sim	26/01/2026	27/01/2026
main_entrance	Portaria	Sim	02/02/2026	02/02/2026
decorative_xmas_tree	Árvore Natalina Decorativa	Não	06/12/2025	05/01/2026

Fonte: Autoria própria, 2026.

Para editar as entidades disponíveis no sistema, existem dois modos distintos: por entidade individual ou por uma lista de objetos JSON — ver Figura 13 (a tela de edição em JSON foi omitida, pois apenas apresenta o mesmo conteúdo de uma forma diferente). O método por JSON facilita a visualização e alteração de várias entidades de uma só vez.

Figura 13 – Nova Entidade - Entidade Individual

UTFPR Virtual

<< Voltar

Slug* example_slug Mesh* Seleccione ☐ Ativo

Posição Eixo X* 1545 Posição Eixo Y* 1545 Posição Eixo Z* 1545

Rotação Eixo X* 1545 Posição Eixo Y* 1545 Posição Eixo Z* 1545

Escala Eixo X* 1545 Escala Eixo Y* 1545 Escala Eixo Z* 1545

Salvar

Editora
editora@utfpr.docentes.com.br

Fonte: Autoria própria, 2026.

A quarta opção de navegação é a lista de arquivos *mesh*, que funciona da mesma forma que os outros módulos - ou seja, é possível visualizar os arquivos já existentes e criar novos (ver Figura 14).

Figura 14 – Lista de Arquivos 3D (Malhas)

UTFPR Virtual

Pesquise pelo nome + Nova Página

Nome	Descrição	Criado em	Modificado em
Bloco Estudantil Genérico	Bloco Estudantil Genérico	01/01/2026	01/01/2026
Portaria	Portaria principal	26/01/2026	27/01/2026
Árvore Genérica 1	Árvore genérica	02/02/2026	02/02/2026
Toldo de cobertura	Toldo de cobertura externa	06/12/2025	05/01/2026

Editora
editora@utfpr.docentes.com.br

Fonte: Autoria própria, 2026.

4.3 Prototipação do banco de dados

É possível ver o banco de dados na íntegra no Apêndice B, todavia, para maior clareza, nesta seção ele é apresentado dividido em 4 fragmentos essenciais — cada um compondo uma parte basilar do projeto:

- **Usuários:** dados referentes aos usuários do sistema;
- **Menu:** itens e categorias do menu de conteúdos (links e outros recursos);
- **Entidades:** entidades que compõe a maquete 3D (prédios, objetos cenográficos ou quaisquer outros elementos);
- **Elementos/conteúdos:** dados que representam os diversos tipos de conteúdos do sistema (textos, vídeos, imagens e diagramas.)

4.3.1 Usuários

A tabela de usuários (ver Figura 15) atua de forma isolada, contendo todas as informações necessárias a respeito do indivíduo e suas permissões, não possuindo chaves estrangeiras vinculadas à quaisquer outras tabelas. Essa decisão fundamenta-se na ideia de que o painel administrativo opera sob uma lógica de governança centralizada. Desta forma, capacidades de auditoria detalhada de adições ou modificações em elementos do sistema não fazem parte do escopo. Desta forma, esta escolha ajuda a manter a estrutura simples e livre de funcionalidades não necessárias.

Analogamente, a diferenciação de privilégios de acesso no sistema adotou uma abordagem simplificado através do atributo *is_admin*. Uma vez que o sistema de permissões opera sob uma lógica binária (editores ou administradores), uma tabela separada para papéis (*roles*) é dispensável.

Figura 15 – Fragmento dos Usuários

users		
id	integer	NN
name	varchar	NN
surname	varchar	NN
email	varchar	NN
password	varchar	NN
token	pb_file	
department	select	NN
id_admin	boolean	NN
created	timestamp	NN
updated	timestamp	NN

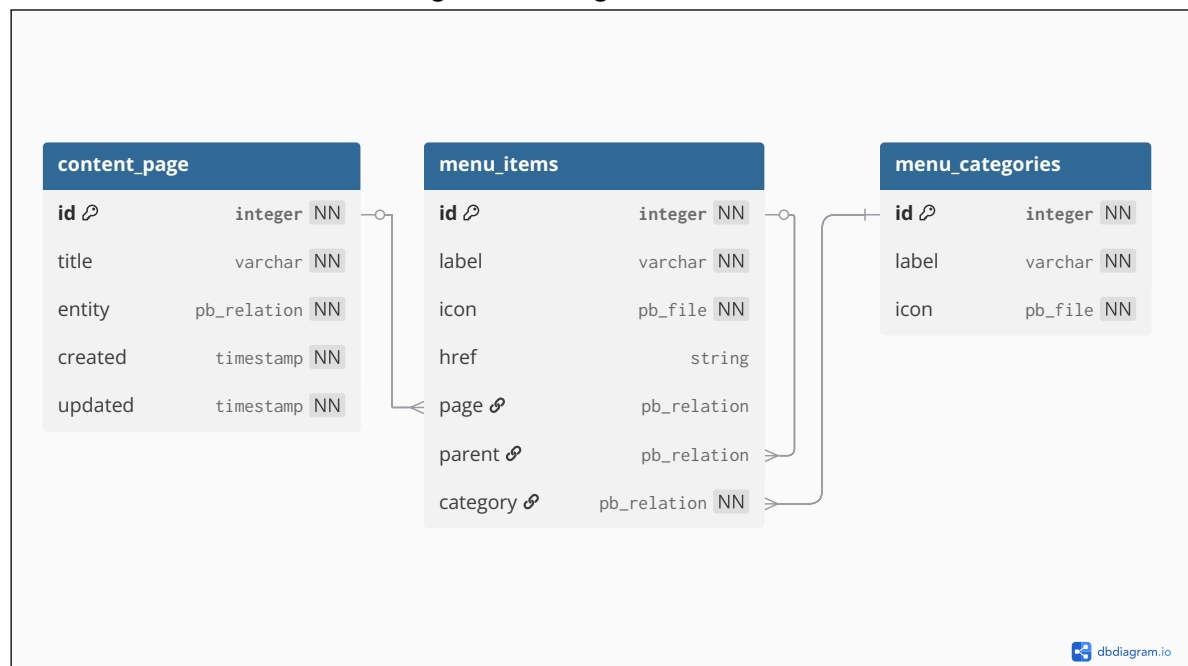
Fonte: Autoria própria (2026).

4.3.2 Menu

Os itens do menu (ver Figura 16) constituem componentes vinculados a uma categoria específica. Além disso, esses itens podem conter dois tipos de ações disparadas pelo clique: a abertura de um link externo ou a exibição de um painel de conteúdo interno (como imagens, textos e diagramas atrelados ao prédio denominado "Bloco B"). Dessa forma, a escolha entre um link e uma página de conteúdo é mutuamente exclusiva, sendo essa uma regra obrigatoriamente reforçada no código da aplicação — tanto nas rotinas de criação dos itens de menu quanto nas ferramentas de validação disponíveis no PocketBase para evitar estados inconsistentes no banco de dados.

Outra característica relevante reside na capacidade dos itens de menu serem agrupados de forma hierárquica. Como exemplo desse comportamento, o item "portal do discente" pode ser configurado de maneira subordinada, posicionando-se diretamente abaixo do item "alunos".

Figura 16 – Fragmento do Menu



Fonte: Autoria própria, 2026.

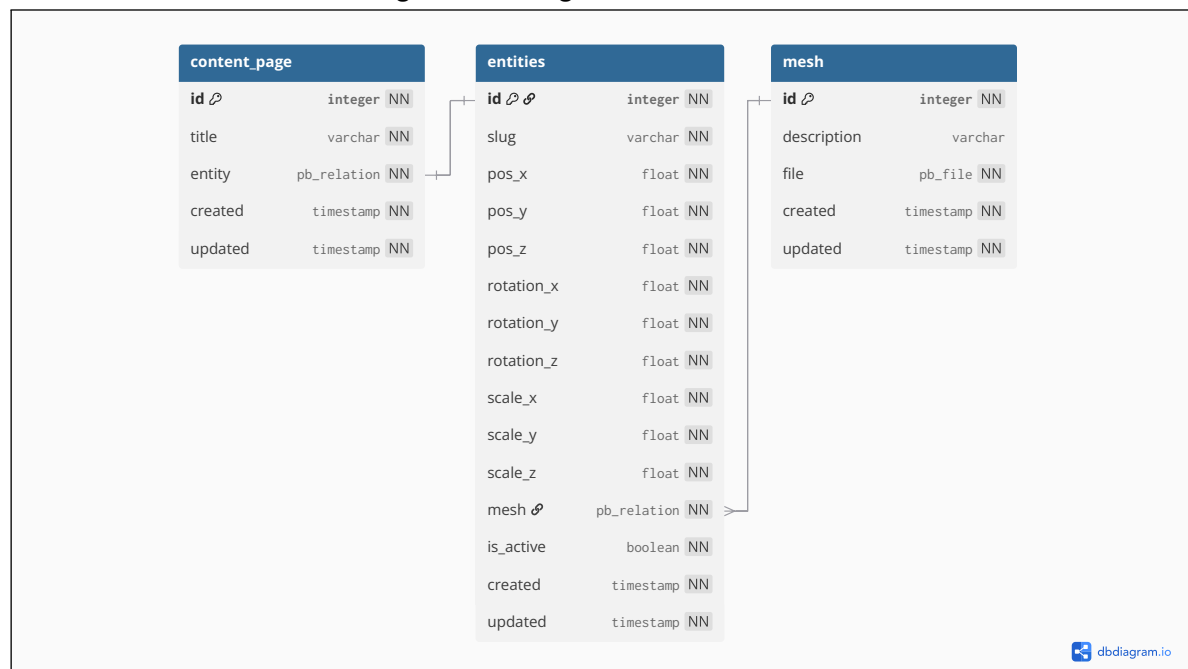
4.3.3 Entidades

As entidades representam todos os objetos contidos na malha tridimensional do campus virtual, abrangendo desde edifícios até elementos cenográficos, tais como postes, árvores e veículos. Cada entidade possui um vínculo exclusivo com apenas um *mesh* — o arquivo tridimensional propriamente dito —, enquanto um mesmo *mesh* pode estar associado a N entidades. No contexto dessa maquete interativa, cada instância possui uma localização espacial única; para rastrear e reproduzir esse posicionamento, o sistema armazena as coordenadas cartesianas nos eixos X, Y e Z, bem como os vetores de rotação e escala correspondentes.

Um ponto adicional de destaque reside no atributo *is_active*, encarregado de sinalizar ao *frontend* quais objetos devem ser renderizados em tempo de execução. Essa abordagem viabiliza cenários em que determinados elementos figuram no cenário apenas sob condições especiais (como, por exemplo, o acionamento de uma árvore natalina comemorativa durante o mês de dezembro).

Por fim, páginas de conteúdo — que agrupam mídias visuais, textos formatados e diagramas dinâmicos — podem ser atreladas de forma direta a uma entidade. O disparo e a exibição desse módulo ocorrem mediante a interação de clique do usuário sobre a geometria tridimensional, sinalizando a associação direta entre a camada de dados textuais e os elementos visuais da cena (ver Figura 17).

Figura 17 – Fragmento das Entidades

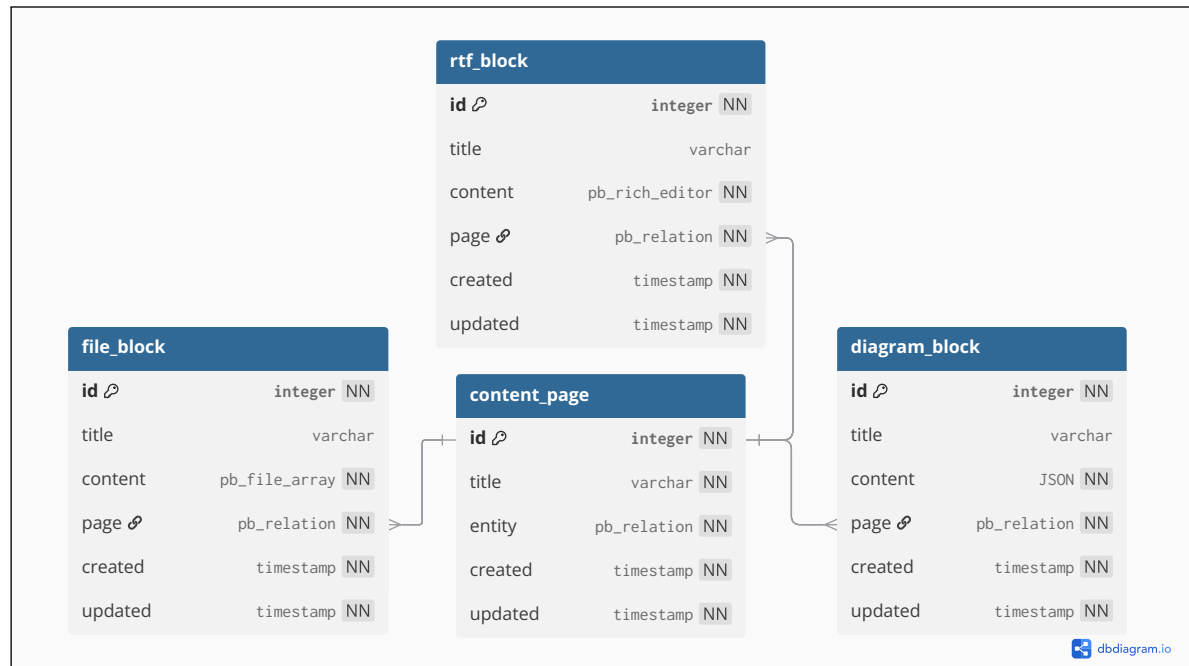


Fonte: Autoria própria, 2026.

4.3.4 Elementos/Conteúdos

Este fragmento representa a modelagem do módulo responsável pelo gerenciamento dinâmico das páginas de conteúdo vinculadas às entidades do campus virtual. Cada página pode abrigar múltiplos blocos informativos, tais como textos formatados, mídias visuais e diagramas. A segregação de cada tipo de componente em tabelas distintas fundamenta-se na necessidade de permitir que uma única página comporte quantos blocos forem necessários, viabilizando a organização modular e a separação desses elementos em categorias (ver Figura 18).

Figura 18 – Fragmento dos Elementos/Conteúdos



Fonte: Autoria própria, 2026.

5 CRONOGRAMA DE TRABALHO

Nesta seção, o cronograma de atividades é apresentado por meio de uma lista detalhada e uma síntese em formato de tabela. A lista descreve a sequência das tarefas, discriminando as etapas de planejamento e fundamentação — compreendidas pelos itens 1 a 5 — que já se encontram concluídas. O detalhamento cronológico é exposto na tabela subsequente, apresentando as projeções de execução para os meses de 2026. Ressalta-se que, no presente momento, as atividades referentes à configuração de infraestrutura, autenticação e a redação da monografia (itens 6, 7 e 15) encontram-se em desenvolvimento, conforme sinalizado na tabela.

1. **Levantamento e Classificação de requisitos:** Análise técnica da estrutura do sistema precursor e reuniões com a ASCOM e a COGETI para levantamento de necessidades e classificação de escopo via método MoSCoW.
2. **Projeto da Arquitetura do Sistema:** Modelagem conceitual da arquitetura cliente-servidor e especificação do barramento de eventos (*event bus*) para o desacoplamento das camadas do *frontend*.
3. **Prototipação das Interfaces de Usuário:** Criação dos protótipos de alta fidelidade e fluxos de navegação no software Figma para validação visual do mapa e do painel administrativo.
4. **Modelagem Lógica do Banco de Dados:** Estruturação lógica das tabelas, definição de tipos de campos e estabelecimento de relacionamentos entre coleções por meio de diagramas lógicos.
5. **Configuração da Infraestrutura e CI/CD:** Estabelecimento do fluxo *gitflow*, integração de ferramentas de checagem estática (ESLint), testes automatizados (Jest/Playwright) e automação de rotinas de implantação via Coolify.
6. **Persistência e Controle de Acesso no Backend:** Implementação prática do esquema de dados no PocketBase e configuração das regras de segurança da API para diferenciação de perfis de acesso.
7. **Módulo de Autenticação e Perfis:** Construção das rotinas de login e validação de sessão no painel administrativo.
8. **Editor de Conteúdo — RTF:** Desenvolvimento estrutural dos blocos de interface voltados à inserção, edição e formatação de RTF.
9. **Editor de Conteúdo — Mídias:** Implementação dos componentes modulares para carregamento, exclusão, tratamento visual e exibição de arquivos de mídia (fotografias e vídeos).

10. **Editor de Conteúdo — Diagramas:** Integração de componentes visuais específicos para a renderização e manipulação de fluxos informacionais cronológicos e linhas do tempo.
11. **Gestão de Entidades e Arquivos de Malha:** Construção das rotinas administrativas para cadastro, edição e vinculação lógica dos metadados informacionais às maquetes tridimensionais.
12. **Modelagem dos Ativos 3D:** Criação das malhas do campus no software Blender, focando na exportação no formato glTF/GLB.
13. **Interface do Mapa Interativo:** Construção do mapa virtual utilizando a biblioteca Three.js, implementando a interatividade da maquete e as barras de ferramentas de navegação.
14. **Testes de Sistema:** Testes de usabilidade por parte da equipe de desenvolvimento e ajustes finais.
15. **Escrita do Relatório Final e Defesa:** Redação contínua dos capítulos do relatório de conclusão, compilação da documentação técnica e preparação para a banca examinadora final.

Tabela 4 – Cronograma de tarefas divididas por meses de 2026

ID	Atividade (Resumo)	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
01	Levantamento e Classificação de Requisitos	X	X										
02	Projeto da Arquitetura do Sistema			X	X								
03	Prototipação das Interfaces de Usuário				X	X							
04	Modelagem Lógica do Banco de Dados					X	X						
05	Configuração da Infraestrutura e CI/CD												
06	Persistência e Controle de Acesso no Backend												
07	Módulo de Autenticação e Perfis												
08	Editor de Conteúdo — RTF												
09	Editor de Conteúdo — Mídias												
10	Editor de Conteúdo — Diagramas												
11	Gestão de Entidades e Malhas												
12	Modelagem dos Ativos 3D												
13	Interface do Mapa Interativo												
14	Testes de Sistema												
15	Escrita do Relatório Final e Defesa	X	X	X	X	X	X						

Fonte: Autoria própria (2026).

6 CONSIDERAÇÕES FINAIS

Este Trabalho de Conclusão de Curso apresenta a reestruturação e modernização do Campus Virtual, propondo uma arquitetura de software e infraestrutura tecnológica renovadas. O ecossistema desenvolvido foi concebido para conferir autonomia aos departamentos administrativos e ao corpo docente, provendo uma ferramenta interativa para a divulgação institucional, gestão de eventos e realização de tours virtuais.

Para além da dinamização no gerenciamento de conteúdo, a solução destaca-se por sua flexibilidade. A estruturação de um núcleo genérico, porém robusto, viabiliza a adaptabilidade do projeto a outros *campi* da instituição, consolidando um modelo técnico capaz de suportar diferentes contextos acadêmicos com esforço de configuração reduzido.

As etapas iniciais deste trabalho foram pautadas pela concepção arquitetural e pelo planejamento estratégico, estabelecendo os alicerces necessários para a viabilidade do sistema. Tais atividades envolveram o uso de metodologias de priorização (MoSCoW), a prototipação de telas e a estruturação da camada de persistência de dados. Superada a fase de design, os esforços agora convergem para o desenvolvimento técnico do MVP. O objetivo primordial desta nova fase é integrar o *frontend* em React com o *backend* em PocketBase, concretizando as ideias projetadas em uma plataforma interativa e funcional que atenda aos objetivos institucionais propostos.

REFERÊNCIAS

ATLASSIAN. **Saiba tudo sobre o Gitflow Workflow**. 2026. Disponível em: <https://www.atlassian.com/br/git/tutorials/comparing-workflows/gitflow-workflow>. Acesso em: 30 de março de 2026.

BERAM, S. **The MoSCoW method for prioritization: A guide for agile teams**. 2023. Disponível em: <https://blog.logrocket.com/product-management/moscow-method-prioritization-agile-examples/>. Acesso em: 30 de março de 2026.

BRASIL. **Portaria CNPq nº 2.664, de 6 de março de 2026. Institui a Política de Integridade na Atividade Científica do CNPq**. 2026. Disponível em: <https://www.in.gov.br/web/dou/-/portaria-cnpq-n-2.664-de-6-de-marco-de-2026-691779232>. Acesso em: 06 de maio de 2026.

Casa do Desenvolvedor. **Requisitos funcionais e não funcionais: o que são e como identificar?** 2023. Disponível em: <https://blog.casadodesenvolvedor.com.br/requisitos-funcionais-e-nao-funcionais/>. Acesso em: 05 de abril de 2026.

CGI.br. **Pesquisa sobre o Uso das Tecnologias de Informação e Comunicação nos Domicílios Brasileiros: TIC Domicílios 2024**. São Paulo, 2025. Disponível em: https://cetic.br/media/docs/publicacoes/2/20250512115624/tic_domicilios_2024_resumo_executivo.pdf. Acesso em: 01 de abril de 2026.

GOOGLE. **Revisão gramatical e refinamento de fluxo de leitura acadêmico**. 2026. Ferramenta: Gemini 1.5 Pro. Prompt: "Atue como revisor acadêmico técnico: corrija gramática, melhore o fluxo de leitura e refine a coesão textual, preservando a terminologia técnica de software e o sentido original do autor.". Disponível em: <https://gemini.google.com>. Acesso em: 06 de maio de 2026.

Open Source Initiative. **The MIT License (MIT)**. 2026. Disponível em: <https://opensource.org/licenses/MIT>. Acesso em: 30 de março de 2026.

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ. **Plano de Desenvolvimento Institucional 2023–2027**. Curitiba, 2023. 302–303 p. Disponível em: <https://nuvem.utfpr.edu.br/index.php/s/rNpmWcJ8plfRQYc#pdfviewer>. Acesso em: 01 de abril de 2026.

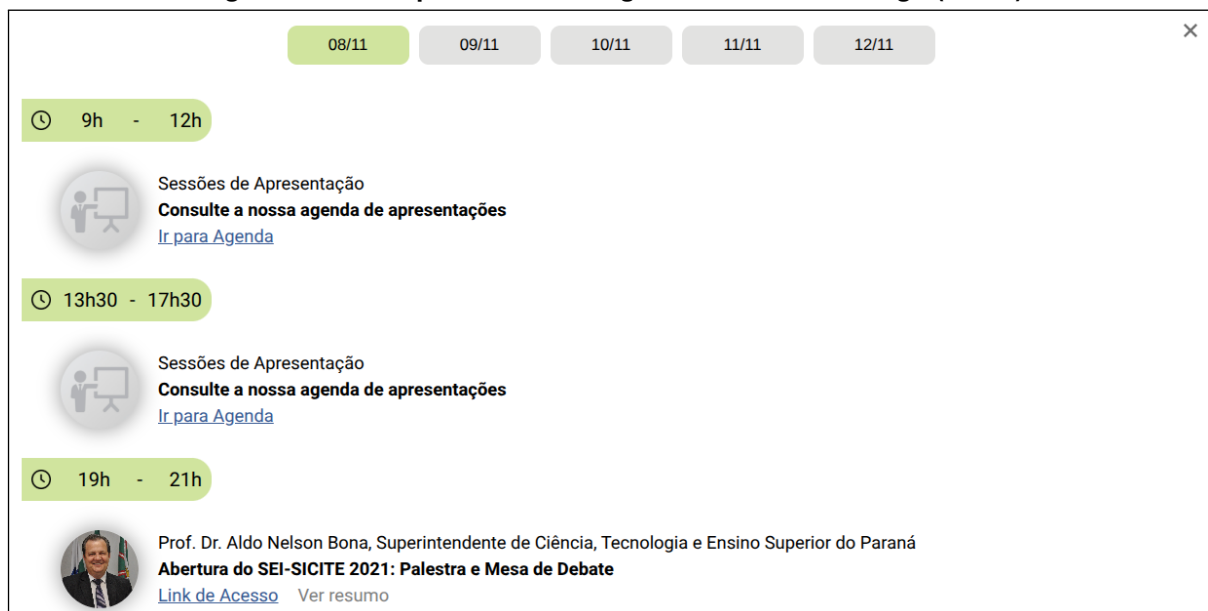
APÊNDICE A – Exemplos da Primeira Versão do Sistema

Figura 19 – Exemplo de um diagrama de linha de tempo no sistema antigo (RF 17)



Fonte: Autoria própria (2026).

Figura 20 – Exemplo de um cronograma no sistema antigo (RF 18)



Fonte: Autoria própria (2026).

APÊNDICE B – Diagrama Completo da Modelagem do Banco de Dados

Figura 21 – Diagrama completo do banco de dados

