



**FACULDADE DE TECNOLOGIA DE PRESIDENTE PRUDENTE  
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**GABRIEL SILVA MANTOVANI**

**GAMIFICANDO O APRENDIZADO DE ENGENHARIA DE SOFTWARE**

**Presidente Prudente – SP**

**2025**



**FACULDADE DE TECNOLOGIA DE PRESIDENTE PRUDENTE  
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**GABRIEL SILVA MANTOVANI**

**GAMIFICANDO O APRENDIZADO DE ENGENHARIA DE SOFTWARE**

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia de Presidente Prudente, como requisito parcial para obtenção do diploma de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Profº. Me. Bruno Santos de Lima

**Presidente Prudente – SP**

**2025**

# **GABRIEL SILVA MANTOVANI**

## **GAMIFICANDO O APRENDIZADO DE ENGENHARIA DE SOFTWARE**

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia de Presidente Prudente, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Aprovado em: 16 de junho de 2025.

### **BANCA EXAMINADORA**

---

Orientador: Profº. Me. Bruno Santos de Lima  
Faculdade de Tecnologia de Presidente Prudente  
Presidente Prudente - SP

---

Profª. Dra. Elaine Parra Affonso  
Faculdade de Tecnologia de Presidente Prudente  
Presidente Prudente - SP

---

Profº. Me. Rodrigo Vilela da Rocha  
Faculdade de Tecnologia de Presidente Prudente  
Presidente Prudente - SP

## **AGRADECIMENTO**

A realização deste trabalho e o percurso da graduação não foram tarefas fáceis, e só foram possíveis graças às pessoas que estão ou já estiveram à minha volta; a todas elas, registro aqui os meus mais sinceros agradecimentos.

Agradeço, primeiramente, ao meu orientador, professor Bruno, que, além das sugestões precisas e das valiosas contribuições, sempre trouxe calma e incentivo nos momentos mais desesperadores causados pelo implacável tempo.

Agradeço, também, aos demais professores, que contribuíram para a minha formação acadêmica, compartilharam experiências importantes e trouxeram oportunidades e visões de mundo únicas.

Agradeço, igualmente, à minha família; afinal, foram longos anos nos quais, felizmente, nunca faltaram apoio, paciência, compreensão e carinho.

Aos amigos que fizeram parte dessa caminhada, dentro e fora da área acadêmica, proporcionaram momentos mais lúdicos e tornaram o trajeto mais prazeroso.

Enfim, foram longos anos, e, sem vocês, eu com certeza não estaria aqui. Meu mais sincero obrigado!

## RESUMO

MANTOVANI, Gabriel. **Gamificando o aprendizado de Engenharia de Software**. Orientador: Bruno Santos de Lima. 2025. Trabalho acadêmico. 56 f. Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistemas) - Faculdade de Tecnologia de Presidente Prudente, Presidente Prudente, SP, 2025.

ProES é uma plataforma de gamificação voltada para a disciplina de Engenharia de Software, alimentada e gerida por professores e estudantes. Seu objetivo principal é oferecer uma alternativa e complemento ao modelo tradicional de ensino, promovendo uma aprendizagem mais dinâmica e envolvente por meio de desafios, recompensas e a aplicação de elementos característicos da gamificação. A proposta surge diante das dificuldades no aprendizado desta disciplina, uma das principais da área, especialmente no que se refere à motivação dos alunos e à compreensão prática de conceitos complexos. A plataforma foi idealizada para permitir que cada unidade de ensino personalize os conteúdos conforme as necessidades específicas de sua grade curricular e de seu público, respeitando assim a individualidade e a autonomia pedagógica dos professores. Dentre as principais funcionalidades, destacam-se os quizzes de múltipla escolha, o sistema de progressão por módulos, conquistas, e moedas virtuais que podem ser trocadas por itens cosméticos, como avatares personalizados. Esses recursos visam não apenas o engajamento, mas também a fixação dos conteúdos. O desenvolvimento do ProES foi realizado com tecnologias web modernas, utilizando framework Laravel e Bootstrap para a estilização. A arquitetura do sistema permite escalabilidade e flexibilidade para futuras melhorias. Espera-se que a utilização do ProES contribua para um aprendizado mais significativo e interativo, aproximando os estudantes dos conteúdos teóricos por meio de atividades práticas e lúdicas. Além disso, o trabalho propõe uma alternativa metodológica para o ensino de Engenharia de Software, servindo como referência para a aplicação de gamificação em outras disciplinas da área de tecnologia.

**Palavras-chave:** gamificação; plataforma de ensino; dinamicidade; aprendizado.

## ABSTRACT

ProES is a gamification platform aimed at the Software Engineering discipline, managed and maintained by teachers and students. Its main objective is to offer an alternative and complement to the traditional teaching model, promoting more dynamic and engaging learning through challenges, rewards, and the application of typical gamification elements. The proposal arises in response to the difficulties in learning this discipline, one of the most important in the field, especially regarding student motivation and the practical understanding of complex concepts. The platform was designed to allow each educational institution to customize the content according to the specific needs of its curriculum and audience, thus respecting the individuality and pedagogical autonomy of teachers. Among the main features are multiple-choice quizzes, a modular progression system, achievements, and virtual coins that can be exchanged for cosmetic items, such as personalized avatars. These resources aim not only to increase engagement but also to enhance content retention. ProES was developed using modern web technologies, leveraging the Laravel framework and Bootstrap for styling. The system's architecture allows for scalability and flexibility for future improvements. It is expected that the use of ProES will contribute to more meaningful and interactive learning, bringing students closer to theoretical content through practical and playful activities. Furthermore, this work proposes a methodological alternative for teaching Software Engineering, serving as a reference for the application of gamification in other disciplines within the field of technology.

**Keywords:** gamification; teaching platform; dynamism; learning.

## **LISTA DE ILUSTRAÇÕES**

Figura 1 -	Layout dos casos de uso Jogar fase e Rejogar fase	16
Figura 2 -	Layout do caso de uso Configurar fases	17
Figura 3 -	Layout do caso de uso Configurar perguntas	17
Figura 4 -	Layout do caso de uso Configurar materiais de apoio	18
Figura 5 -	Layout do caso de uso Comprar cosméticos	18
Figura 6 -	Diagrama de Casos de Uso	19
Figura 7 -	Diagrama de Atividade Jogar fase	22
Figura 8 -	Diagrama de Atividade Rejogar fase	23
Figura 9 -	Diagrama de Atividade Configurar fases	23
Figura 10 -	Diagrama de Atividade Configurar perguntas	24
Figura 11 -	Diagrama de Atividade Configurar materiais de apoio	24
Figura 12 -	Diagrama de Atividade Comprar cosméticos	25
Figura 13 -	Modelo Conceitual	25
Figura 14 -	Diagrama de Sequência do caso de uso Jogar fase	26
Figura 15 -	Diagrama de Sequência do caso de uso Rejogar fase	27
Figura 16 -	Diagrama de Sequência do caso de uso Configurar fase	28
Figura 17 -	Diagrama de Sequência do caso de uso Configurar pergunta	29
Figura 18 -	Diagrama de Sequência do caso de uso Configurar material de apoio	30
Figura 19 -	Diagrama de Sequência do caso de uso Comprar cosmético	31
Figura 20 -	Diagrama de Classes	32
Figura 21 -	Modelagem da Base de Dados	33
Figura 22 -	Diagrama de Pacotes da Arquitetura Lógica	34
Figura 23 -	Modelo Navegacional para usuários do tipo aluno	34
Figura 24 -	Modelo Navegacional para usuários do tipo professor	35

Figura 25 - Layout do Dashboard	35
Figura 26 - Layout do placar de líderes	36
Figura 27 - Layout da exibição dos módulos	36
Figura 28 - Layout do conteúdo dos módulos	37
Figura 29 - Layout da coleção do usuário	37
Figura 30 - Layout das conquistas do usuário	38
Figura 31 - Layout da exibição do perfil de um usuário	38

## **LISTA DE ABREVIATURAS E SIGLAS**

ENADE      Exame Nacional de Desempenho de Estudantes

INEP      Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira

## SUMÁRIO

<b>1. INTRODUÇÃO.....</b>	<b>5</b>
1.1 OBJETIVO.....	5
1.2 ESCOPO.....	5
1.3 DEFINIÇÕES, SIGLAS E ABREVIAÇÕES.....	6
1.4 REFERÊNCIAS.....	9
1.5 VISÃO GERAL.....	10
<b>2. DESCRIÇÃO GERAL DO PRODUTO.....</b>	<b>10</b>
2.1 ESTUDO DE VIABILIDADE.....	11
2.2 FUNÇÕES DO SISTEMA.....	12
2.3 CARACTERÍSTICAS DO USUÁRIO.....	14
2.4 LIMITES, DEPENDÊNCIAS E SUPOSIÇÕES.....	14
2.5 REQUISITOS ADIADOS.....	15
<b>3. REQUISITOS ESPECÍFICOS.....</b>	<b>15</b>
3.1 Requisitos de Interface Externa.....	16
3.1.1 Interfaces do Usuário dos Casos de Uso.....	16
3.1.2 Interfaces de Software.....	19
3.2 DIAGRAMA DE CASOS DE USO.....	19
3.3 ESPECIFICAÇÕES DOS CASOS DE USO.....	19
3.4 DIAGRAMAS DE ATIVIDADES DOS CASOS DE USO.....	22
3.5 MODELO CONCEITUAL.....	25
4. PROJETO DE SOFTWARE.....	26
4.1 DIAGRAMAS DE INTERAÇÃO.....	26
4.2 DIAGRAMA DE CLASSES.....	32
4.3 MODELAGEM DA BASE DE DADOS.....	33
4.4 DIAGRAMA DE PACOTES DA ARQUITETURA LÓGICA.....	34
4.5 MODELO NAVEGACIONAL.....	34
4.6 OUTROS LAYOUTS DE TELAS.....	35
<b>APÊNDICE A – Procedimentos para Implantação do Sistema.....</b>	<b>38</b>
<b>ANEXO 1 – Referências.....</b>	<b>41</b>
<b>ANEXO 2 – Manual do Usuário.....</b>	<b>1</b>

## 1. INTRODUÇÃO

A gamificação tem ganhado cada vez mais espaço em diversas áreas, sendo utilizada como uma poderosa ferramenta de engajamento e motivação. No contexto educacional, essa estratégia tem sido adotada para aumentar a participação ativa dos alunos, facilitar a assimilação de conteúdos e estimular o desenvolvimento de competências. A Engenharia de Software, disciplina fundamental na formação de profissionais de tecnologia, apresenta desafios específicos devido à sua natureza técnica e abstrata, exigindo abordagens inovadoras para tornar o aprendizado mais dinâmico e envolvente.

Neste projeto, propõe-se a criação de uma ferramenta de gamificação voltada para o ensino da Engenharia de Software. A ferramenta tem por objetivo auxiliar no processo de ensino-aprendizagem, abordando temas como modelos de processos de software, especificação de requisitos e gerenciamento de projetos. Ao integrar elementos lúdicos e interativos ao ensino, espera-se aumentar o engajamento dos estudantes ou ao menos possibilitar que tenham uma melhor absorção do conteúdo trabalhado, contribuindo para o desenvolvimento das habilidades necessárias para sua formação.

### 1.1 OBJETIVO

Este documento consiste em uma *ERS* (Especificação de Requisitos de Software) baseada na norma *IEEE 830/1998* (Institute of Electrical and Electronics Engineers) e tem como objetivo especificar os requisitos do software em desenvolvimento, integrando o cliente e os desenvolvedores sobre o desenvolvimento e a utilização do software.

### 1.2 ESCOPO

ProES é uma plataforma de gamificação que tem por objetivo auxiliar no ensino da Engenharia de Software, para que seja uma tarefa mais lúdica e proveitosa, que contemple outras formas de aprender.

As funções básicas do sistema abrangem o gerenciamento das contas pessoais e o visual do personagem.

Como uma plataforma de gamificação, o sistema tem como funcionalidade fundamental a jogabilidade, e iniciar uma fase/desafio é o ponto principal, permitindo que o jogador possa escolher dentre as fases disponíveis uma que queira praticar.

Para que não fique defasado com o tempo, o sistema também permite a inclusão de novos desafios, sempre com um desenvolvimento e aprovação prévia dos envolvidos.

A fim de reforçar o aprendizado, a revisão das fases, ou seja, a possibilidade de jogá-las novamente também é uma função fundamental no sistema, desta vez com mais informações devido à última experiência do usuário para que sirva de motivação para obter resultados melhores.

O sistema permite disponibilizar nas fases questões reais de provas advindas de concursos, métricas de cursos e afins, para tentar trazer um pouco da experiência para um aluno que tenha este objetivo.

O material didático disponível nos módulos poderá ser disponibilizado pelos professores quando bem entenderem e acharem necessário, até mesmo atualizar um antigo.

Ainda na perspectiva das funções fundamentais, será possível comprar modificações para o personagem, como uma forma de recompensar os esforços nos desafios.

Como funções de saída podem ser consultadas as pontuações gerais dos jogadores, o próprio progresso, os perfis dos usuários, as opções de cosméticos na loja e as próprias conquistas.

Por fim, a ideia principal do ProES é facilitar e ser eficiente em transmitir o conhecimento, por isso a dinamicidade no compartilhamento dos conteúdos e desafios, e das conquistas como algo a ser alcançado de maneira lúdica e recompensadora.

### 1.3 DEFINIÇÕES, SIGLAS E ABREVIACÕES

**Gamificação:** é a aplicação das estratégias dos jogos nas atividades do dia a dia, com o objetivo de aumentar o engajamento dos participantes. Ela se baseia no game thinking, conceito que abrange a integração da gamificação com outros saberes do meio corporativo e do design.

**Engenharia de Software (ES):** disciplina que se preocupa com o desenvolvimento sistemático, disciplinado e mensurável de software.

**Modelos de Processo de Software:** abordagens metodológicas para o desenvolvimento de software, como o Modelo Cascata, Scrum, e Kanban.

**ERS (Especificação de Requisitos de Software):** documento que descreve as funcionalidades e restrições de um sistema de software a ser desenvolvido.

**F\_F:** funções fundamentais, que indicam maior importância para o sistema.

**F\_B:** funções básicas, que indicam papéis ou questões a serem gerenciadas ou mantidas.

**F\_S:** funções de saída, que indicam as exibições do sistema para as consultas dos usuários.

**PHP:** linguagem de programação amplamente usada para desenvolvimento web, especialmente para aplicações no lado do servidor (back-end).

**Back-end:** parte da aplicação responsável pela lógica de negócios, processamento de dados e comunicação com bancos de dados, normalmente executada no servidor.

**Front-end:** parte da aplicação responsável pela interface e interação com o usuário.

**Framework:** conjunto de ferramentas, bibliotecas e boas práticas que facilitam e padronizam o desenvolvimento de aplicações.

**Laravel:** framework PHP moderno e popular para desenvolvimento web, que segue o padrão MVC e oferece recursos para simplificar tarefas comuns, como autenticação, roteamento e ORM.

**Blade:** motor de templates nativo do Laravel que permite criar interfaces HTML dinâmicas utilizando uma sintaxe simples e elegante.

**Breeze:** kit inicial oficial do Laravel que fornece uma implementação simples e minimalista de autenticação, utilizando Blade ou outras stacks de front-end.

**Bootstrap:** framework front-end para desenvolvimento de interfaces responsivas, utilizando componentes e classes CSS e JavaScript predefinidas.

**SO (Sistema Operacional):** software que gerencia o hardware do computador e fornece serviços para a execução de programas, como Windows, Linux ou macOS.

**MySQL:** Sistema de Gerenciamento de Banco de Dados Relacional (SGBDR) amplamente utilizado, que armazena dados em tabelas organizadas.

**Apache:** servidor web gratuito e open source, responsável por processar requisições HTTP e entregar conteúdos web aos navegadores.

**IDE:** ambiente de desenvolvimento integrado que reúne editor de código, depurador, compilador e outros recursos para facilitar o desenvolvimento de software. Ex.: Visual Studio Code, PhpStorm.

**Git:** sistema de controle de versão distribuído que permite gerenciar o histórico de alterações de código, colaborando com equipes e facilitando o versionamento.

**MVC (Model-View-Controller):** padrão de arquitetura de software que separa uma aplicação em três camadas: Model (dados), View (interface) e Controller (lógica).

**Open source:** software de código aberto, cujo código-fonte está disponível publicamente, permitindo estudo, modificação e distribuição por qualquer pessoa.

**LGPD (Lei Geral de Proteção de Dados):** legislação brasileira que regula o tratamento de dados pessoais, visando garantir a privacidade e segurança das informações.

**HTTPS:** versão segura do protocolo HTTP, que utiliza criptografia SSL/TLS para proteger os dados transmitidos entre servidor e cliente.

**Open SSL:** é uma biblioteca open source que implementa protocolos de segurança como SSL/TLS.

**Let's encrypt:** autoridade certificadora gratuita, automatizada e open source que oferece certificados SSL/TLS para proteção de sites via HTTPS.

**PDO (PHP Data Objects):** extensão do PHP que fornece uma interface consistente para acesso a bancos de dados, permitindo trabalhar com múltiplos SGBDs usando a mesma API.

**Mbstring:** extensão do PHP que fornece suporte para operações com strings multibyte, essencial para trabalhar com textos em diferentes codificações, como UTF-8.

**Tokenizer:** extensão do PHP que analisa código fonte, transformando-o em tokens, útil para interpretadores, compiladores ou análise de código.

**XML (eXtensible Markup Language):** linguagem de marcação para armazenar e transportar dados estruturados de forma legível para humanos e máquinas.

**Ctype:** extensão do PHP que fornece funções para checar caracteres de strings, como verificar se são dígitos, letras ou outros tipos.

**JSON (JavaScript Object Notation):** formato leve e amplamente usado para troca de dados estruturados entre sistemas, baseado em texto e facilmente legível.

**Composer:** gerenciador de dependências para PHP, que facilita a instalação e atualização de bibliotecas e frameworks.

**Node.js:** ambiente de execução JavaScript no lado do servidor, que permite criar aplicações de rede escaláveis e rápidas.

**NPM (Node Package Manager):** gerenciador de pacotes oficial do Node.js, usado para instalar bibliotecas e ferramentas JavaScript.

**Yarn:** alternativa ao NPM, é um gerenciador de pacotes JavaScript focado em velocidade, segurança e consistência.

**.env:** arquivo de configuração utilizado para armazenar variáveis de ambiente, como senhas, chaves de API e configurações específicas da aplicação, sem expor esses dados no código.

**JavaScript:** linguagem de programação interpretada, essencial para o desenvolvimento web, que permite criar interatividade e dinamismo em páginas web.

**Digital Ocean:** provedor de infraestrutura em nuvem que oferece serviços como servidores virtuais, armazenamento e banco de dados gerenciado.

**AWS (Amazon Web Services):** plataforma de computação em nuvem da Amazon, com uma ampla gama de serviços, como servidores, bancos de dados, armazenamento e inteligência artificial.

**Heroku:** plataforma como serviço (PaaS) que permite o deploy de aplicações web de forma simplificada, gerenciando automaticamente infraestrutura e escalabilidade.

#### 1.4 REFERÊNCIAS

Os documentos relacionados no *Quadro 1* encontram-se no *Anexo 1*.

**Quadro 1** – Referências adquiridas.

Nº	Título	Data de aquisição	Responsável pelo fornecimento
----	--------	-------------------	-------------------------------

1	ENADE 2008 Prova de Tecnologia em Análise e Desenvolvimento de Sistemas	19/05/25	INEP
2	ENADE 2011 Tecnologia em Análise e Desenvolvimento de Sistemas	19/05/25	INEP
3	ENADE 2014 Tecnologia em Análise e Desenvolvimento de Sistemas	19/05/25	INEP
4	ENADE 2017 Prova de Tecnologia em Análise e Desenvolvimento de Sistemas	19/05/25	INEP
5	ENADE 2021 Prova de Tecnologia em Análise e Desenvolvimento de Sistemas	19/05/25	INEP

Fonte: Elaborado pelo próprio autor.

## 1.5 VISÃO GERAL

Além deste capítulo de Introdução, que apresentou o objetivo da ERS, escopo do produto, definições e referências. Esta *ERS* está organizada em demais capítulos. O *Capítulo 2* fornece uma descrição geral do software em desenvolvimento, contendo uma perspectiva do produto, suas funções, perfil dos usuários do software, características do desenvolvimento e requisitos adiados. O *Capítulo 3* detalha os requisitos do software. O *Capítulo 4* fornece detalhes do projeto do software.

## 2. DESCRIÇÃO GERAL DO PRODUTO

### 2.1 ESTUDO DE VIABILIDADE

O sistema ProES será desenvolvido como uma plataforma web, a fim de garantir fácil acesso e ampla compatibilidade entre dispositivos, sem necessidade de instalação local. A aplicação será implementada utilizando a linguagem de programação PHP, por meio do framework Laravel. A interface web será construída com o Blade e estilizada com Bootstrap. O sistema poderá ser executado em servidores web com sistemas operacionais Linux ou Windows, sendo acessível via navegadores modernos em sistemas Windows, macOS e Linux.

Para o funcionamento da plataforma, será necessário um servidor de hospedagem com configurações básicas adequadas para rodar aplicações web desenvolvidas em Laravel, como um ambiente com suporte a PHP, MySQL e servidor web (Apache, por exemplo). As estações de trabalho dos professores e dos alunos precisarão apenas de um dispositivo com acesso à internet e um navegador atualizado. Para manutenção da plataforma poderão ser utilizadas IDEs Visual Studio Code ou alguma outra que atenda às linguagens e para o versionamento do código ferramentas como o Git.

Descrição	Quantidade	Situação	Preço unitário	Preço total
Servidor de hospedagem	1	Não possui	R\$ 50,00/mês	R\$ 600,00/ano
Sistema operacional (Linux)	1	Já possui (open source)	R\$ 0,00	R\$ 0,00
Banco de Dados MySQL	1	Já possui (open source)	R\$ 0,00	R\$ 0,00
IDE (VS Code)	1	Já possui (open source)	R\$ 0,00	R\$ 0,00
Domínio .com.br	1	Não possui	R\$ 65,00/ano	R\$ 65,00/ano
Certificado SSL	1	Não possui	R\$ 0,00 (Let's encrypt)	R\$ 0,00

A escolha por uma solução web oferece diversas vantagens, tais como a facilidade de acesso de qualquer local com conexão à internet, sem necessidade de instalação prévia. Além disso, a utilização de tecnologias open source reduz significativamente os custos de licenciamento de software. A escolha do Laravel proporciona um desenvolvimento estruturado e seguro, com uma ampla comunidade de suporte. A arquitetura baseada em MVC facilita a manutenção e evolução do sistema. Por fim, a escalabilidade é garantida por meio da possibilidade de hospedagem em serviços de nuvem, ajustando recursos conforme a demanda.

O desenvolvimento e a utilização do sistema ProES deverão estar em conformidade com a Lei Geral de Proteção de Dados (LGPD - Lei nº 13.709/2018), principalmente devido ao tratamento de dados pessoais de professores e alunos, como nomes, emails e instituições de ensino. Dessa forma, é imprescindível a implementação de práticas de segurança da informação, como o uso de criptografia em senhas e conexões seguras via HTTPS.

## 2.2 FUNÇÕES DO SISTEMA

O Quadro 2 apresenta as funções básicas do sistema, ou seja, as operações CRUD.

**Quadro 2 – Funções Básicas.**

Identificação	Descrição
F_B01 Gerenciar usuários	Permite incluir, excluir, alterar e pesquisar contas dos estudantes e professores da plataforma.
F_B02 Gerenciar personagem	Permite alterar o próprio personagem usado na plataforma.
F_B03 Gerenciar conquistas	Permite consultar as conquistas.
F_B04 Gerenciar módulos de fases	Permite alterar e pesquisar módulos das fases.

Fonte: Elaborado pelo próprio autor.

O Quadro 3 apresenta as funções fundamentais do sistema, ou seja, as implementações das regras de negócio.

**Quadro 3 – Funções Fundamentais.**

<b>Identificação</b>	<b>Descrição</b>
F_F01 Jogar fase	O sistema permitirá que o estudante selecione e inicie uma fase de aprendizado. Essa funcionalidade será acessível por meio de uma interface intuitiva onde o jogador poderá escolher entre diferentes fases que abordam conteúdos específicos, de acordo com seu nível de progresso na plataforma. Ao iniciar uma fase, o conteúdo será apresentado de maneira sequencial, mas não obrigatória.
F_F02 Rejogar fase	O estudante poderá revisar fases já concluídas, acessando novamente o conteúdo para reforçar os conceitos aprendidos. A revisão será feita de forma interativa, com a possibilidade de refazer exercícios e desafios relacionados à fase, permitindo uma melhor retenção do conhecimento e a consolidação das habilidades adquiridas ao longo do jogo.
F_F03 Configurar fase	O professor poderá gerenciar os conteúdos da plataforma, de maneira que permita constante atualização e expansão dos desafios.
F_F04 Configurar pergunta	O professor poderá gerenciar as perguntas dos desafios, seja criando novas ou alterando as já existentes.
F_F05 Configurar material de apoio	Ao professor será possível incluir materiais didáticos nos módulos da própria plataforma para que agreguem no conhecimento dos estudantes que os buscarem.
F_F06 Comprar cosméticos	A plataforma disponibilizará uma loja de cosméticos para que os usuários possam escolher acessórios para customizar seus avatares, a moeda do jogo será uma das recompensas das fases.
F_F07 Realizar exercício	O jogador poderá responder às questões propostas nas fases usando os meios disponibilizados pela interface.
F_F08 Atualizar pontuação	É uma função fundamental oculta que ocorre quando é jogada uma fase e respondida uma pergunta ou exercício, só é de fato atualizada se as respostas forem enviadas.
F_F09 Receber conquista	Função fundamental oculta que acontece ao jogar uma fase ou ao alcançar algum feito no sistema e assim uma conquista pré configurada será habilitada para o perfil em questão.
F_F10 Atualizar moedas	Também uma função fundamental oculta que acontece ao finalizar uma fase ou ao gastá-las na loja.
F_F11 Equipar cosméticos	A plataforma disponibilizará uma página um inventário para que os usuários possam equipar seus acessórios comprados.

Fonte: Elaborado pelo próprio autor.

O Quadro 4 apresenta as funções de saída do sistema, ou seja, relatórios, gráficos e listagens.

**Quadro 4 – Funções de Saída.**

Identificação	Descrição
F_S01 Relatório geral da pontuação dos jogadores	O relatório será exibido aos professores para que possam acompanhar o andamento e progresso dos alunos nas fases, isso, de uma maneira geral, auxilia o professor a entender como vai o andamento dos alunos para com as fases.
F_S02 Barra de progresso	O jogador/estudante poderá consultar sua barra de progresso nas fases, a qual demonstrará quantas fases já foram concluídas comparadas às existentes
F_S03 Visualizar desempenho da fase recém finalizada	O aluno saberá de imediato se concluiu a fase e quantas questões acertou por meio do feedback das moedas ganhas e de um marcador na própria fase.
F_S04 Visualizar o perfil dos jogadores	O professor, por enquanto, poderá visualizar o perfil dos jogadores a partir do relatório geral da pontuação dos jogadores.
F_S05 Visualizar conquistas	O estudante ou professor poderá visualizar as conquistas alcançadas e não alcançadas e saber como obtiveram ou poderão obtê-las.
F_S06 Visualizar cosméticos	O jogador poderá visualizar as opções para seu personagem na loja ou na coleção antes de equipá-lo, ou então nele mesmo quando o tiver e usar.
F_S07 Verificar últimos feitos	O jogador saberá qual a última fase jogada e a última conquista alcançada ao entrar na plataforma.

Fonte: Elaborado pelo próprio autor.

### 2.3 CARACTERÍSTICAS DO USUÁRIO

Os usuários da plataforma, idealmente, ou são professores ou estudantes de um curso da área da tecnologia da informação, visto que a disciplina tratada é a de Engenharia de Software, podendo estar no ensino técnico ou superior.

### 2.4 LIMITES, DEPENDÊNCIAS E SUPOSIÇÕES

O sistema depende de uma hospedagem para o servidor, e para os usuários uma conexão com a internet estável e um computador com acesso a navegadores da atualidade.

## 2.5 REQUISITOS ADIADOS

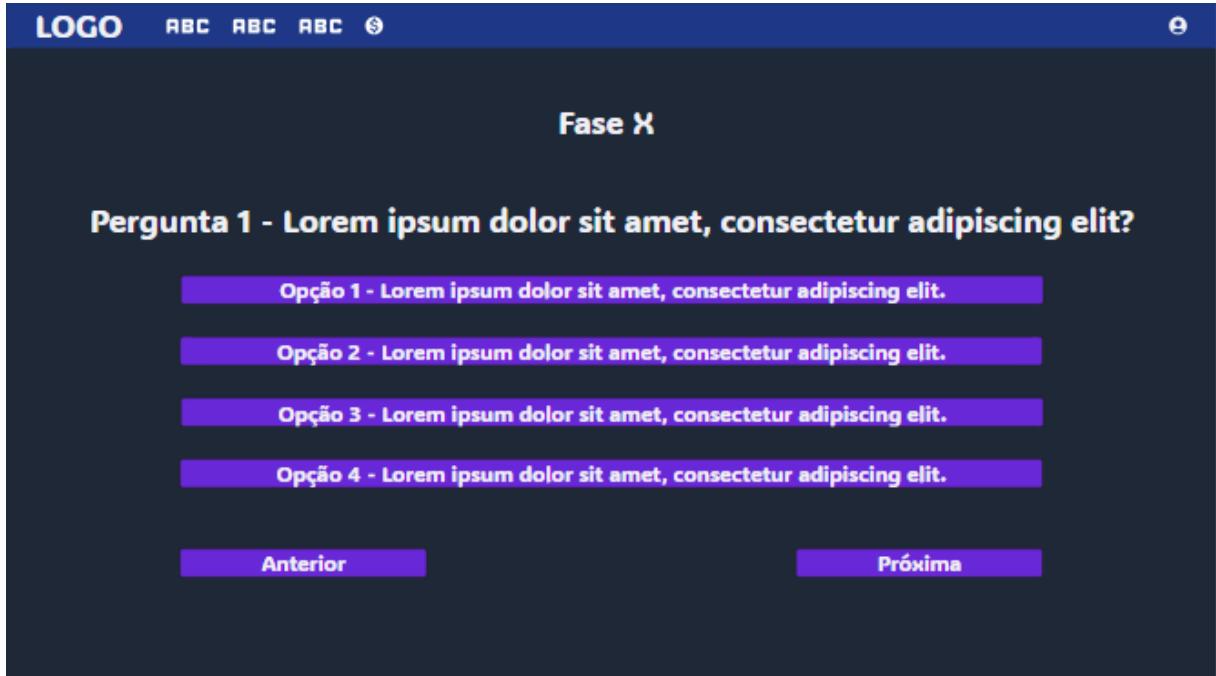
O requisito *F\_B03* será melhorado em versões futuras do site, permitindo que pessoas autorizadas possam criar conquistas e regras para obtê-las.

### 3. REQUISITOS ESPECÍFICOS

#### 3.1 Requisitos de Interface Externa

##### 3.1.1 Interfaces do Usuário dos Casos de Uso

Figura 1 – Layout dos casos de uso Jogar fase e Rejogar fase.



Fonte: Elaborado pelo próprio autor.

**Figura 2 – Layout do caso de uso Gerenciar fases.**

**Título da fase**

**Pergunta**

**Respostas**

**Adicionar resposta**

**Adicionar pergunta**

**Salvar**

Fonte: Elaborado pelo próprio autor.

**Figura 3 – Layout do caso de uso Gerenciar perguntas.**

**Título da fase**

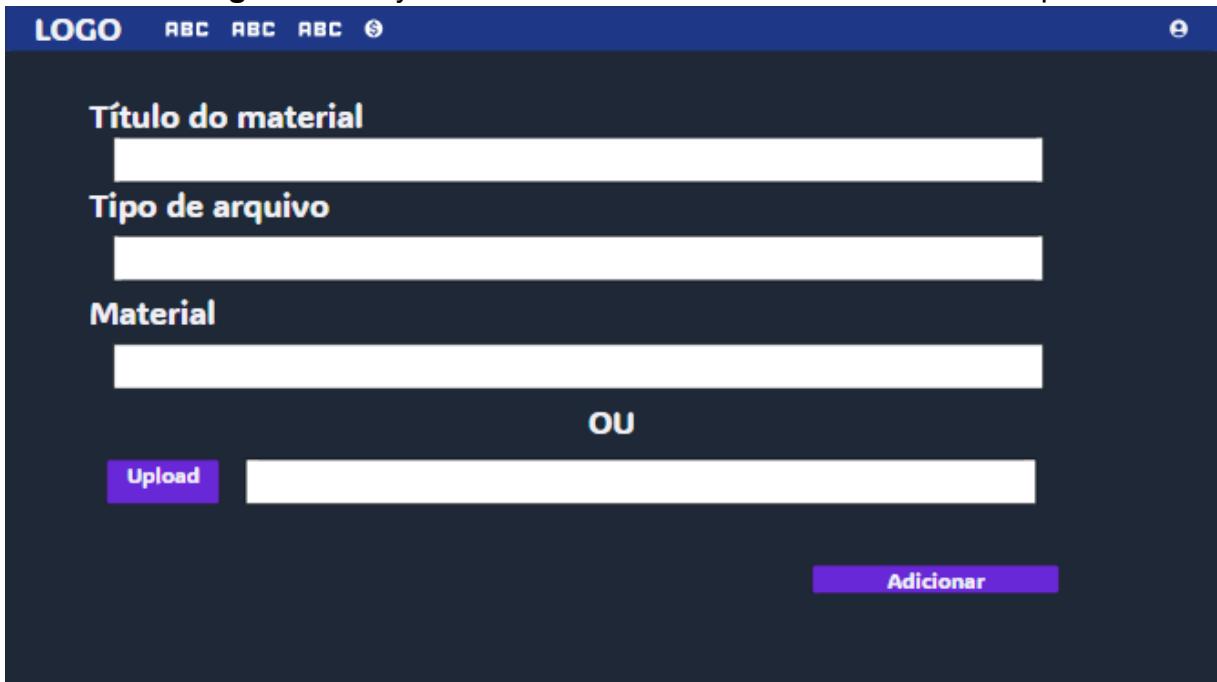
**Pergunta**

**Respostas**

**Salvar**

Fonte: Elaborado pelo próprio autor.

**Figura 4 – Layout do caso de uso Gerenciar materiais de apoio.**



Fonte: Elaborado pelo próprio autor.

**Figura 5 – Layout do caso de uso Comprar cosméticos.**



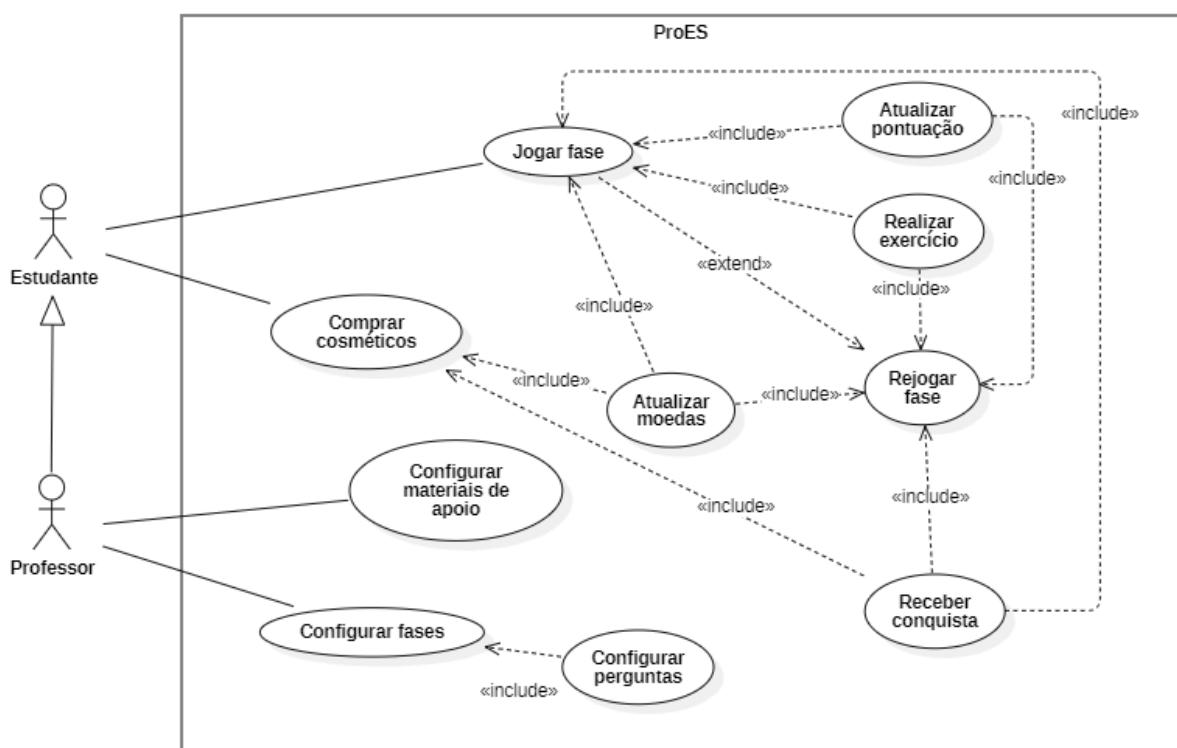
Fonte: Elaborado pelo próprio autor.

### 3.1.2 Interfaces de Software

O uso de um Sistema Gerenciador de Banco de Dados é imprescindível para o administrador, visto que algumas permissões só podem ser dadas por eles, e o sistema operacional deve possibilitar acesso à internet e a um navegador.

## 3.2 DIAGRAMA DE CASOS DE USO

**Figura 6** – Diagrama de Casos de Uso.



Fonte: Elaborado pelo próprio autor.

## 3.3 ESPECIFICAÇÕES DOS CASOS DE USO

### 3.3.1 Jogar fase

**Ator principal:** Estudante ou professor.

**Pré condições:** Ter concluído o módulo anterior, se houver.

**Pós condições:** Fase concluída ou não, se sim e última do módulo desbloqueia o próximo, se houver, a menos que seja a última. Quantidade de moedas as quais a fase poderá entregar para o mesmo usuário atualizada.

**Cenário de sucesso principal:**

1. O estudante inicia uma das fases do módulo.
2. O sistema exibe os desafios da fase.
3. O estudante realiza os exercícios até a conclusão e envia as respostas.

4. O sistema registra as respostas e faz a correção das perguntas, calculando as recompensas.
5. Após o cálculo das recompensas, o sistema acrescenta a pontuação ao saldo das moedas do jogador.
6. O sistema verifica se alguma conquista foi alcançada, se sim a habilita, se não nada é feito a respeito.
7. O sistema exibe as recompensas atingidas.
8. O estudante visualiza sua recompensa pela fase.

**Cenário alternativo:**

- 3a. O estudante interage com a fase, todavia não a conclui. O sistema não contabiliza a jogada e finaliza o caso de uso.

**3.3.2 Rejogar fase**

**Ator principal:** Estudante ou professor.

**Pré condições:** A fase já deve ter sido jogada.

**Pós condições:** Quantidade de moedas as quais a fase poderá entregar para o mesmo usuário atualizada..

**Cenário de sucesso principal:**

1. O estudante inicia uma das fases do módulo.
2. O sistema exibe os desafios da fase.
3. O estudante realiza os exercícios até a conclusão e envia as respostas.
4. O sistema registra as respostas e faz a correção das perguntas, calculando as recompensas, agora comparando com a melhor pontuação da fase, pois há um limite de moedas que podem ser obtidas por cada fase igual ao número de perguntas da mesma.
5. Após o cálculo das recompensas, o sistema acrescenta a pontuação ao saldo das moedas do jogador.
6. O sistema verifica se alguma conquista foi alcançada, se sim a habilita, se não nada é feito a respeito.
7. O sistema exibe as recompensas atingidas.
8. O estudante visualiza sua recompensa pela fase.

**Cenário alternativo:**

- 3a. O estudante interage com a fase, todavia não a conclui. O sistema não contabiliza a jogada e finaliza o caso de uso.

**3.3.3 Configurar fase**

**Ator principal:** Professor.

**Pré condições:** Se alteração ou exclusão a fase deve ter sido inserida pelo próprio professor.

**Pós condições:** Fase criada, alterada ou excluída.

**Cenário de sucesso principal:**

1. O professor navega até o módulo desejado e escolhe com qual das opções irá interagir com a fase de sua autoria.
2. O sistema responde de acordo com a opção, seja ela criação, alteração ou exclusão da fase. Se a opção é de exclusão o caso de uso termina aqui exibindo a mensagem de sucesso ao usuário. Se alteração e o usuário for

alterar as perguntas ou respostas vai para o caso de uso “Configurar pergunta”. Já se a opção é a criação, o caso continua.

3. Ao professor optar por criar uma nova fase, o sistema exibe as informações que precisa e disponibiliza ao professor quantas perguntas e respostas ele quiser incrementar até que adicione a fase ao módulo.
4. O sistema guarda a fase no módulo e exibe uma mensagem de sucesso.

### 3.3.4 Configurar pergunta

**Ator principal:** Professor.

**Pré condições:** Haver uma fase já criada pelo professor.

**Pós condições:** Pergunta inserida, atualizada ou excluída.

**Cenário de sucesso principal:**

1. O professor seleciona uma fase a qual quer fazer alguma alteração.
2. O sistema exibe suas informações e deixa que o professor decida o que quer fazer.
3. O professor altera o que julgar necessário e salva as alterações.
4. O sistema valida as novas informações e exibe uma mensagem de sucesso.

**Cenário alternativo:**

- 4a. O sistema não aceita os dados da validação e exibe uma mensagem de erro.

### 3.3.5 Configurar material de apoio

**Ator principal:** Professor.

**Pré condições:** Se alteração ou exclusão o material deve ter sido inserido pelo próprio professor.

**Pós condições:** O material é incluído, atualizado ou excluído do módulo.

**Cenário de sucesso principal:**

1. O professor navega até o módulo desejado e escolhe com qual das opções irá interagir com o material de sua autoria.
2. O sistema responde de acordo com a opção escolhida. Se selecionada a exclusão, o caso de uso acaba aqui, se alteração ou criação o caso continua com o sistema exibindo a tela adequada para tal situação.
3. O professor faz as alterações ou inserções de dados e salva o procedimento.
4. O sistema valida e aceita o material e exibe uma mensagem de sucesso.

**Cenário alternativo:**

- 4a. O sistema valida e não aceita o material e exibe uma mensagem de erro.

### 3.3.6 Comprar cosméticos

**Ator principal:** Estudante ou professor.

**Pré condições:** O usuário não tem o cosmético que quer comprar e tem saldo suficiente.

**Pós condições:** O usuário adquire um novo cosmético para usar.

**Cenário de sucesso principal:**

1. O usuário vai até a loja.
2. O sistema exibe na loja as opções disponíveis.
3. O usuário escolhe uma delas e tenta comprá-la.

4. O sistema verifica que o saldo de moedas do usuário é suficiente, atualiza as moedas do mesmo e habilita o cosmético para o usuário.
5. O sistema verifica se alguma conquista foi alcançada, se sim a habilita, se não nada é feito a respeito.
6. A personalização passa a ficar disponível na coleção do usuário.

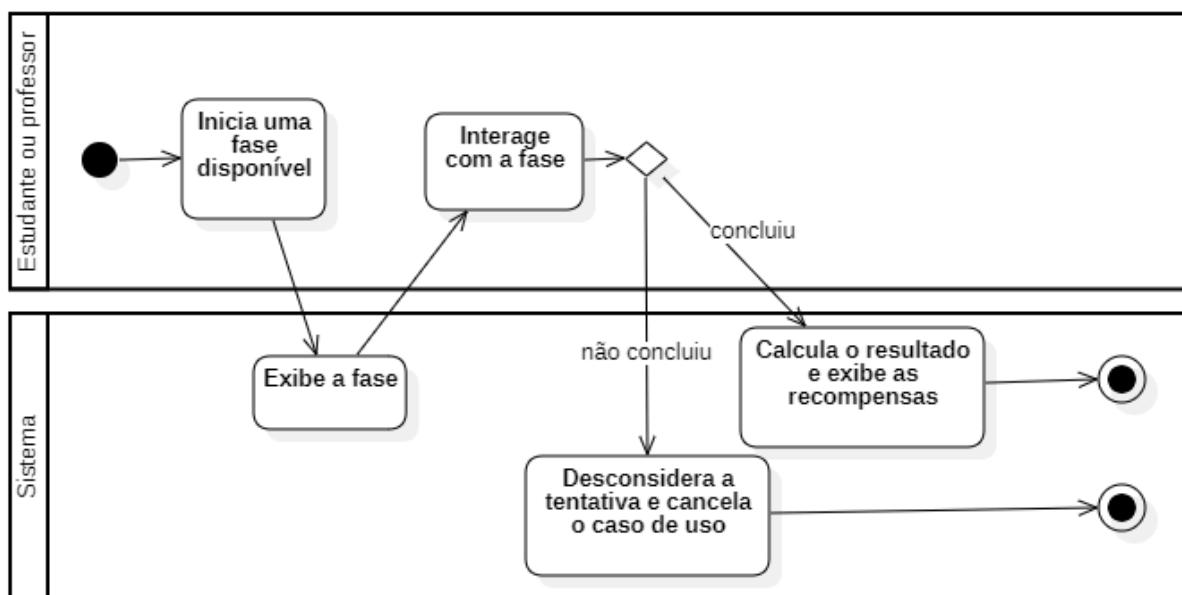
**Cenário alternativo:**

- 4a. O usuário escolhe um item que não tem moedas suficientes para comprar e o sistema percebe não permitindo a compra e notificando o usuário sobre o ocorrido voltando a exibir a loja.

### 3.4 DIAGRAMAS DE ATIVIDADES DOS CASOS DE USO

#### 3.4.1 Jogar fase

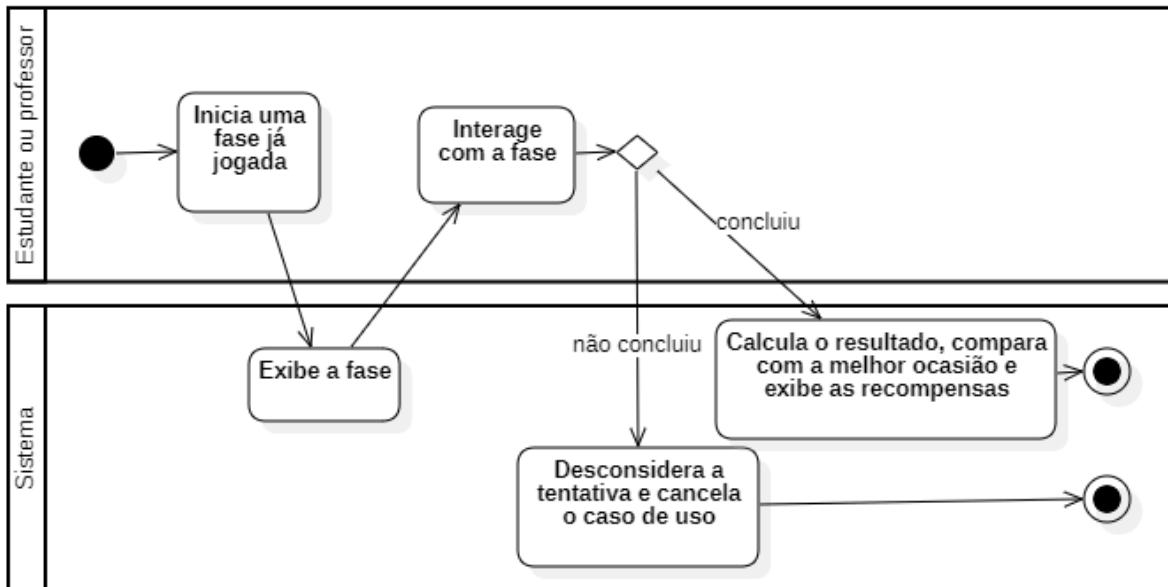
**Figura 7 – Diagrama de Atividade Jogar fase.**



Fonte: Elaborado pelo próprio autor.

### 3.4.2 Rejogar fase

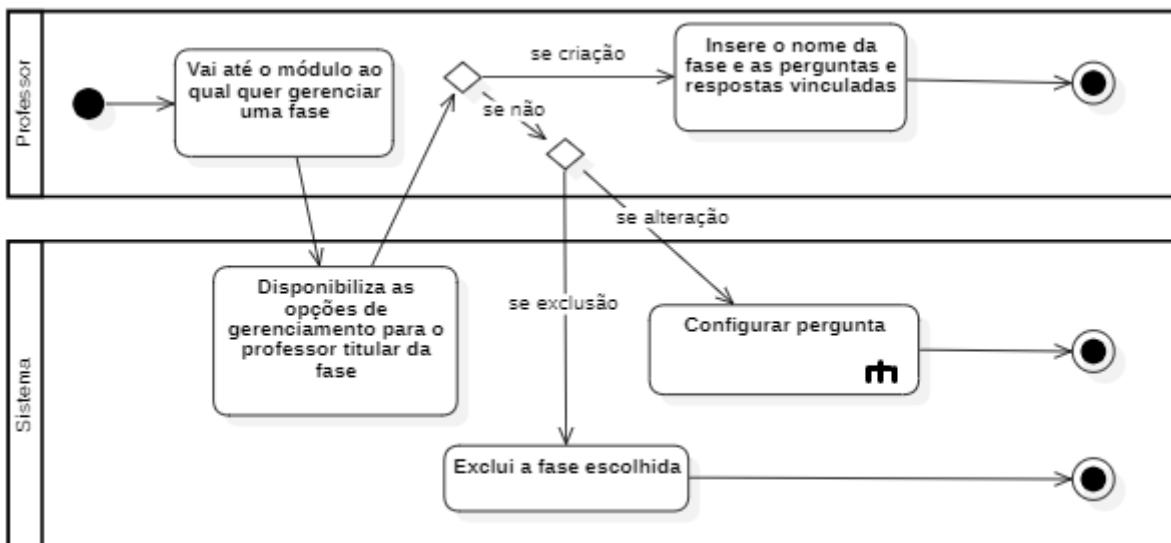
**Figura 8** – Diagrama de Atividade Rejogar fase.



Fonte: Elaborado pelo próprio autor.

### 3.4.3 Configurar fase

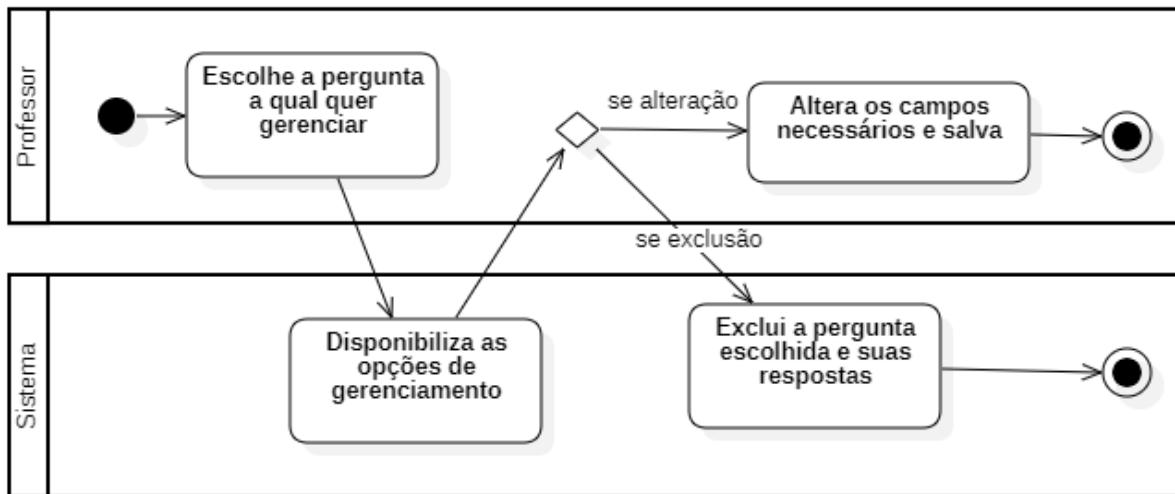
**Figura 9** – Diagrama de Atividade Configurar fase.



Fonte: Elaborado pelo próprio autor.

### 3.4.4 Configurar pergunta

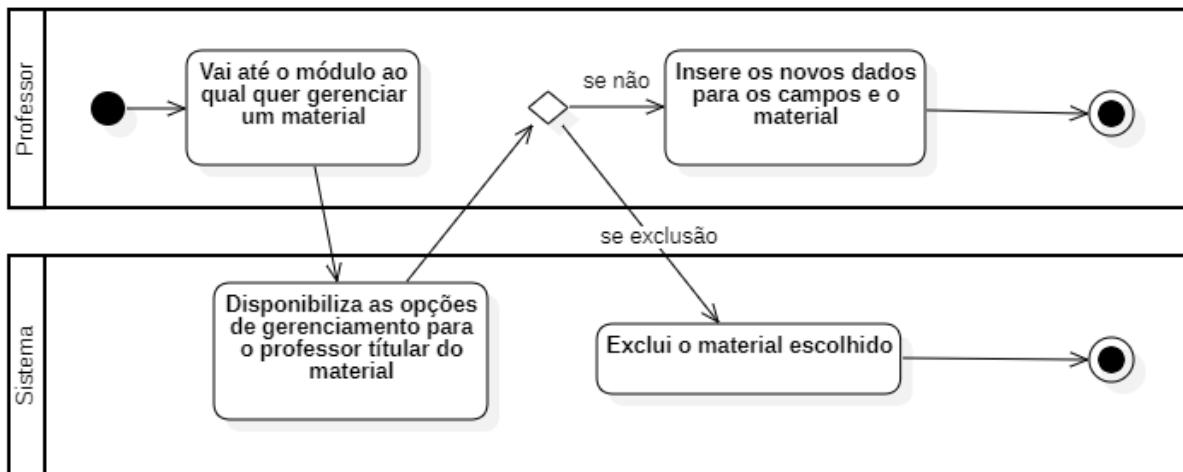
**Figura 10** – Diagrama de Atividade Configurar pergunta.



Fonte: Elaborado pelo próprio autor.

### 3.4.5 Configurar material de apoio

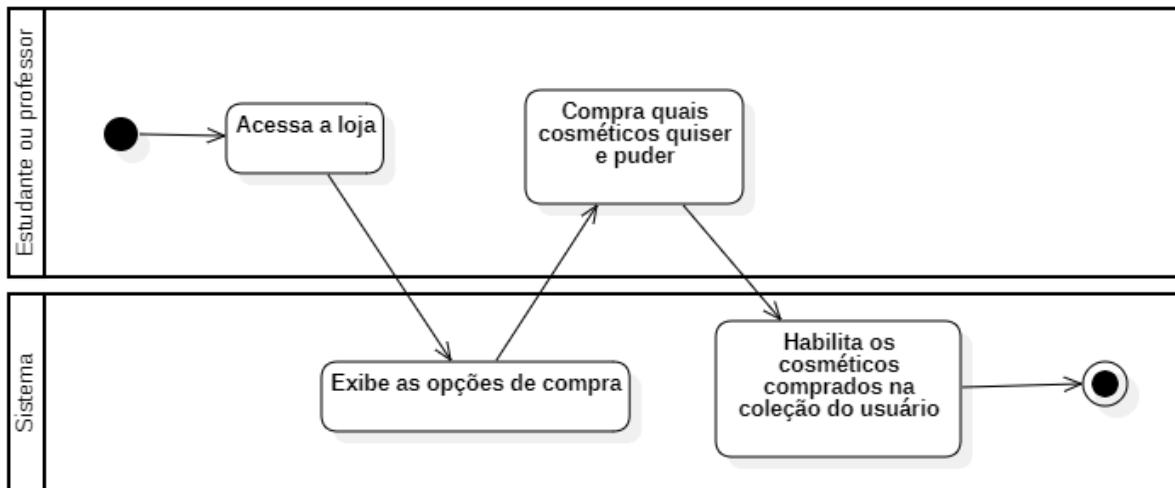
**Figura 11** – Diagrama de Atividade Configurar material de apoio.



Fonte: Elaborado pelo próprio autor.

### 3.4.6 Comprar cosméticos

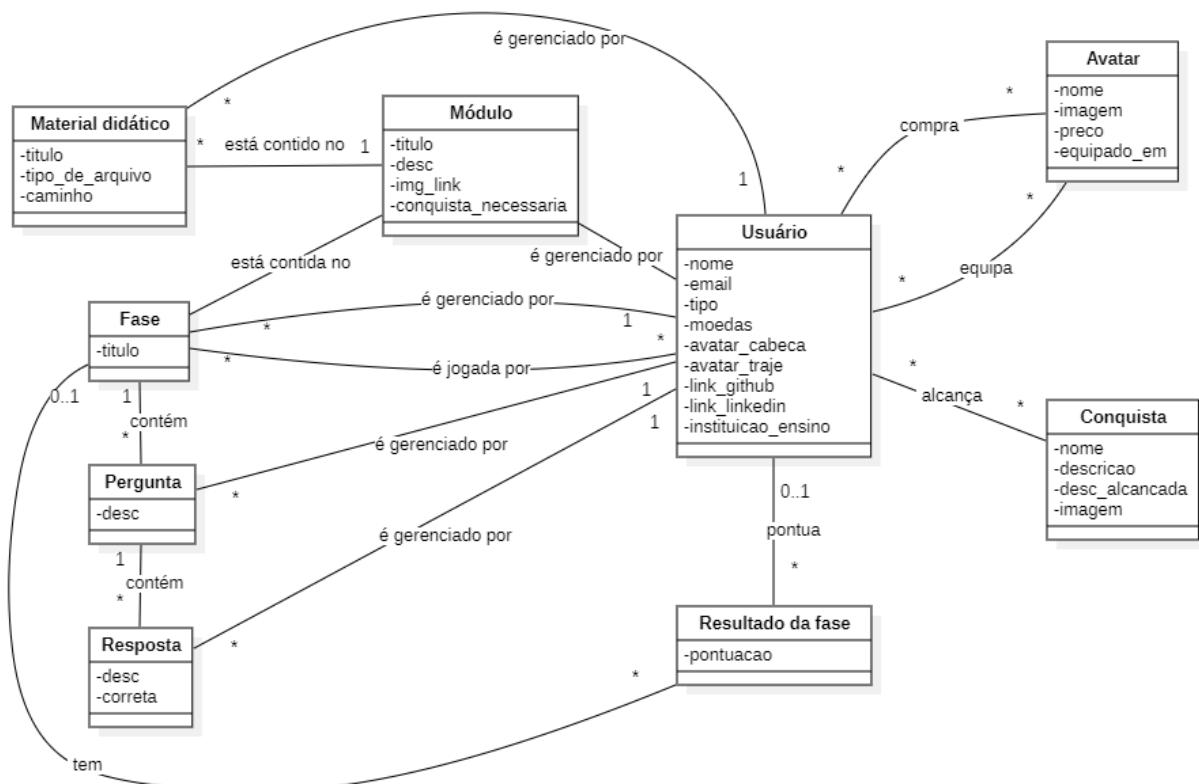
**Figura 12** – Diagrama de Atividade Comprar cosméticos.



Fonte: Elaborado pelo próprio autor.

### 3.5 MODELO CONCEITUAL

**Figura 13** – Modelo Conceitual.

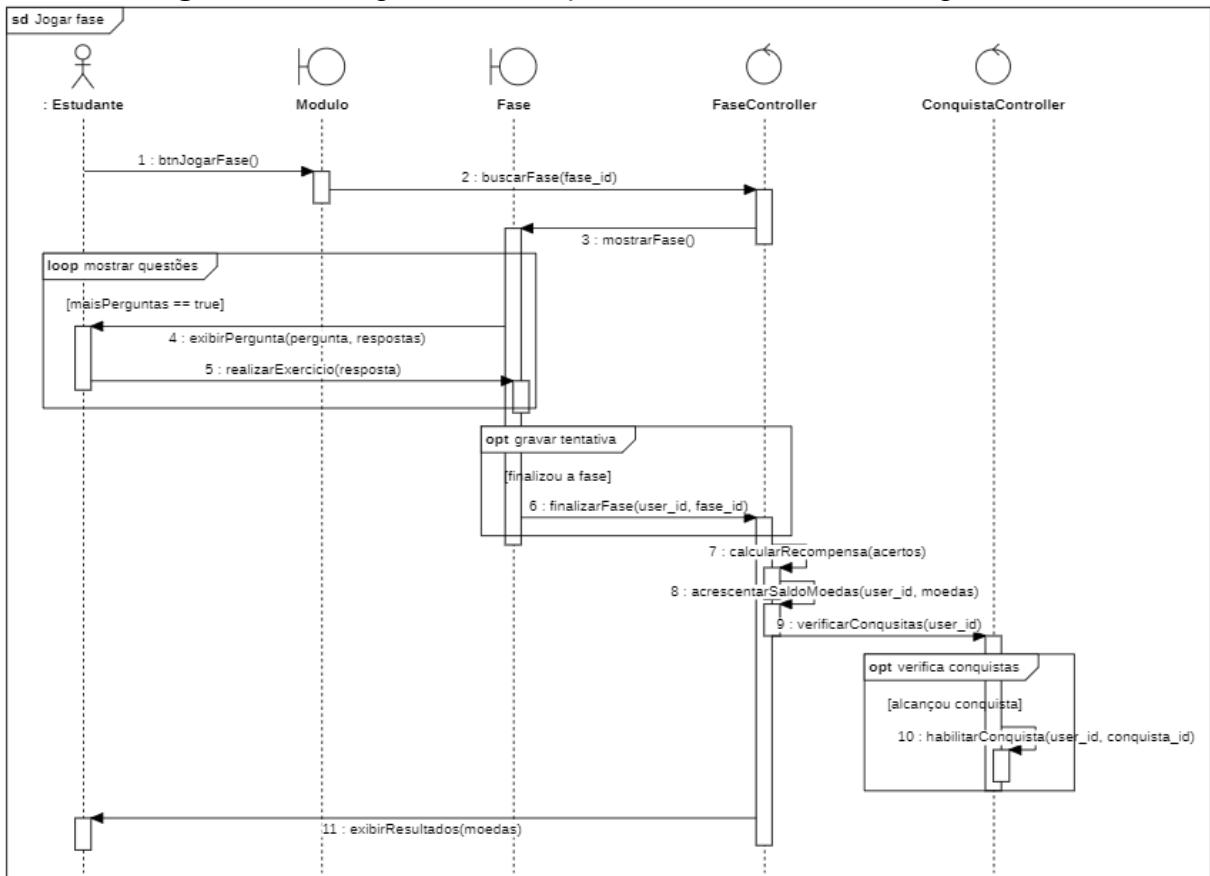


Fonte: Elaborado pelo próprio autor.

## 4. PROJETO DE SOFTWARE

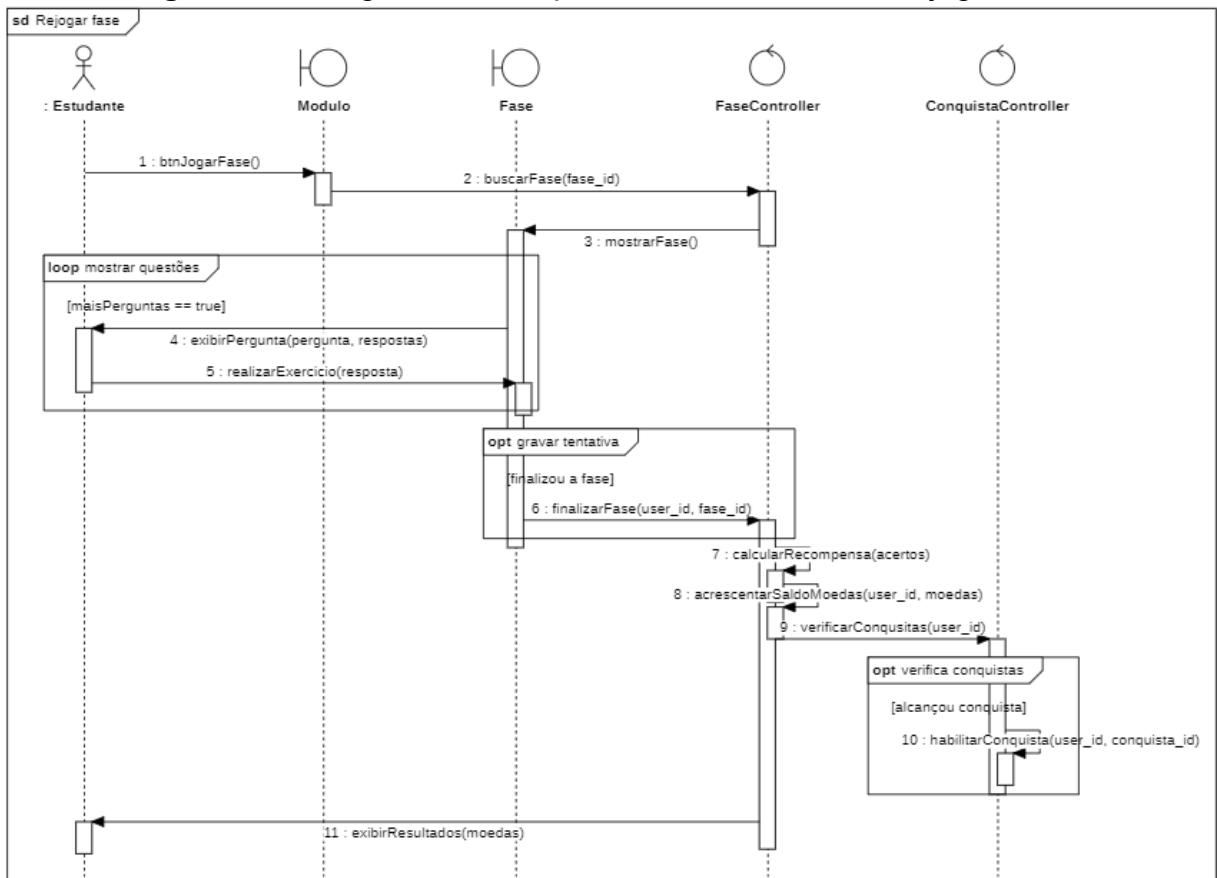
### 4.1 DIAGRAMAS DE INTERAÇÃO

**Figura 14** – Diagrama de Sequência do caso de uso Jogar fase.



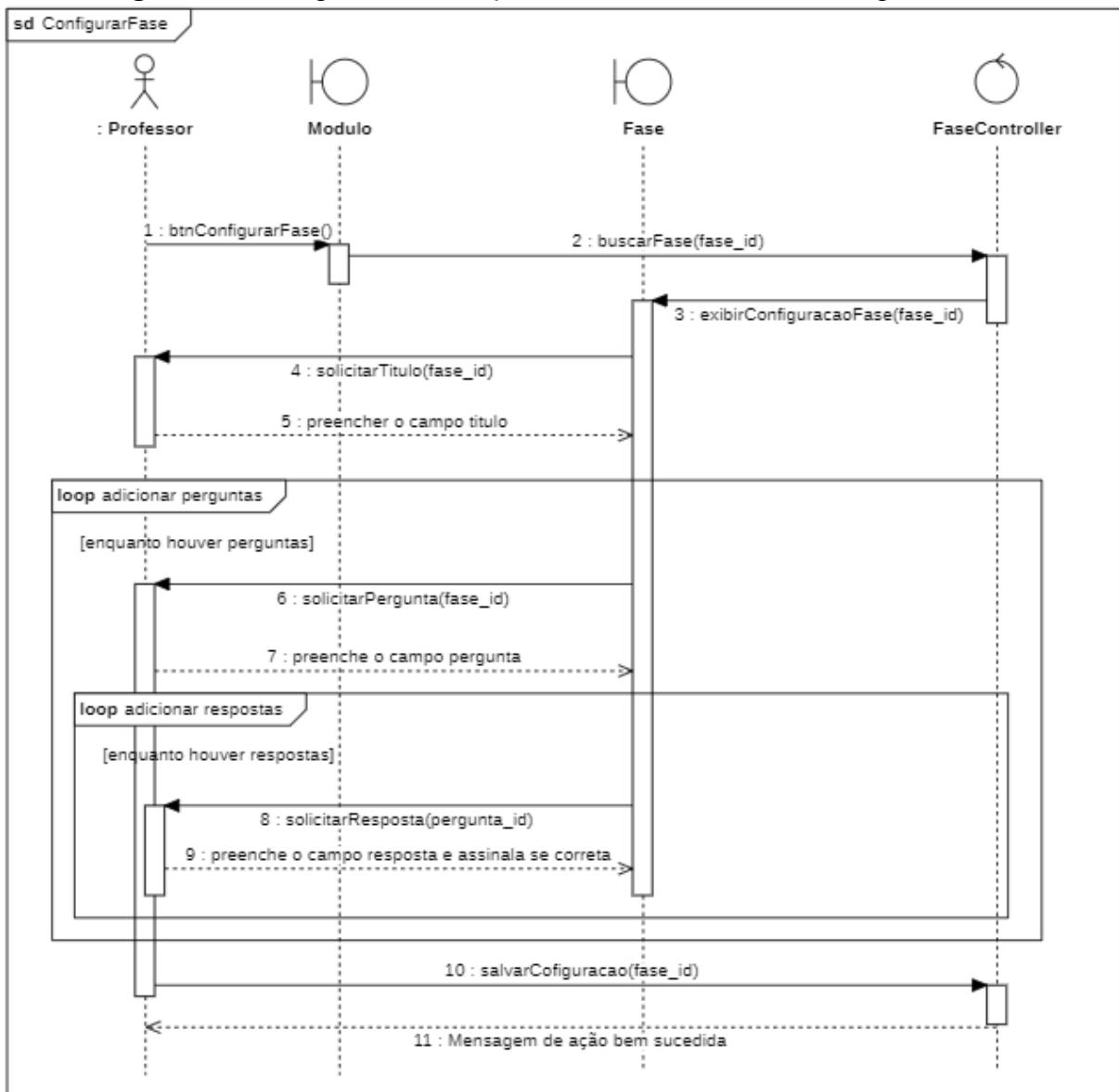
Fonte: Elaborado pelo próprio autor.

**Figura 15 – Diagrama de Sequência do caso de uso Rejogar fase.**



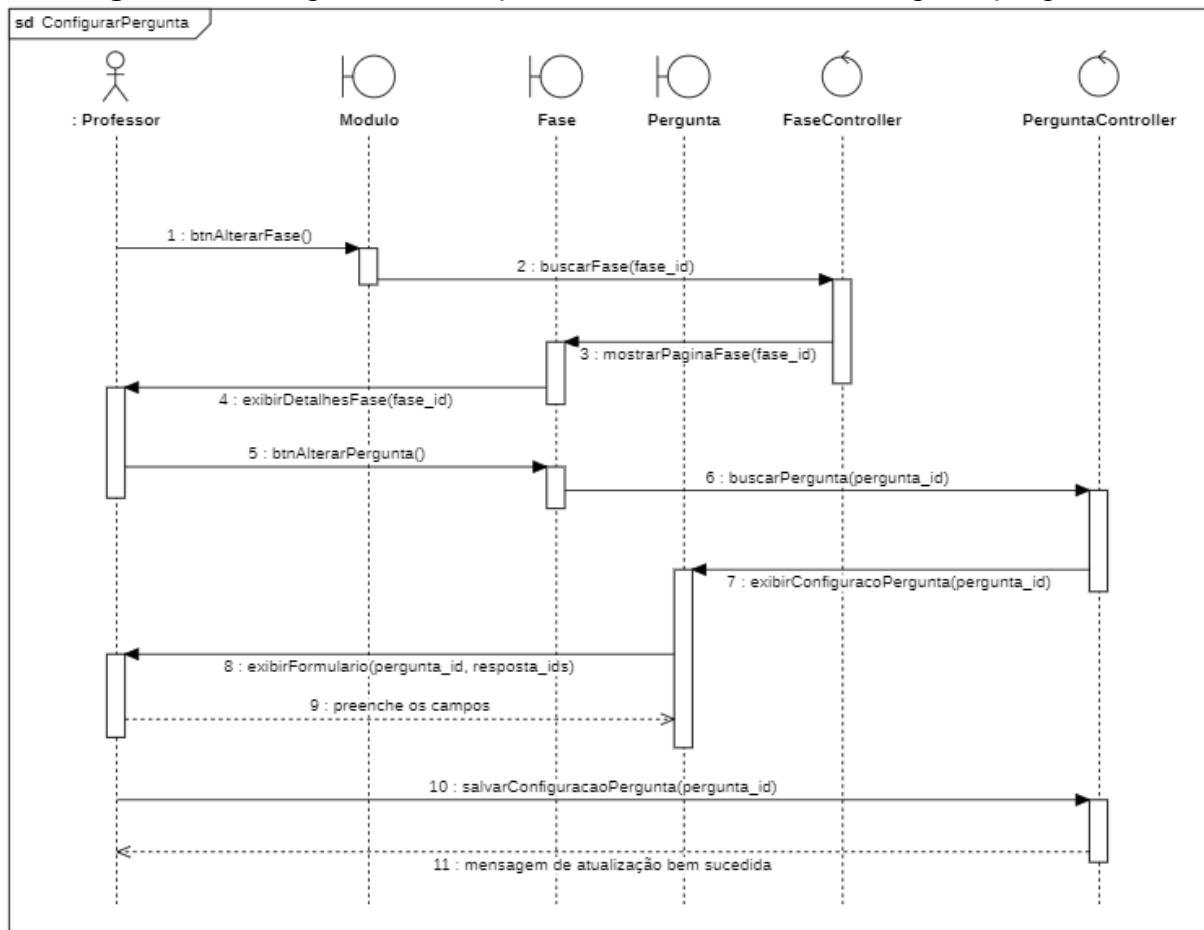
Fonte: Elaborado pelo próprio autor.

**Figura 16 – Diagrama de Sequência do caso de uso Configurar fase.**



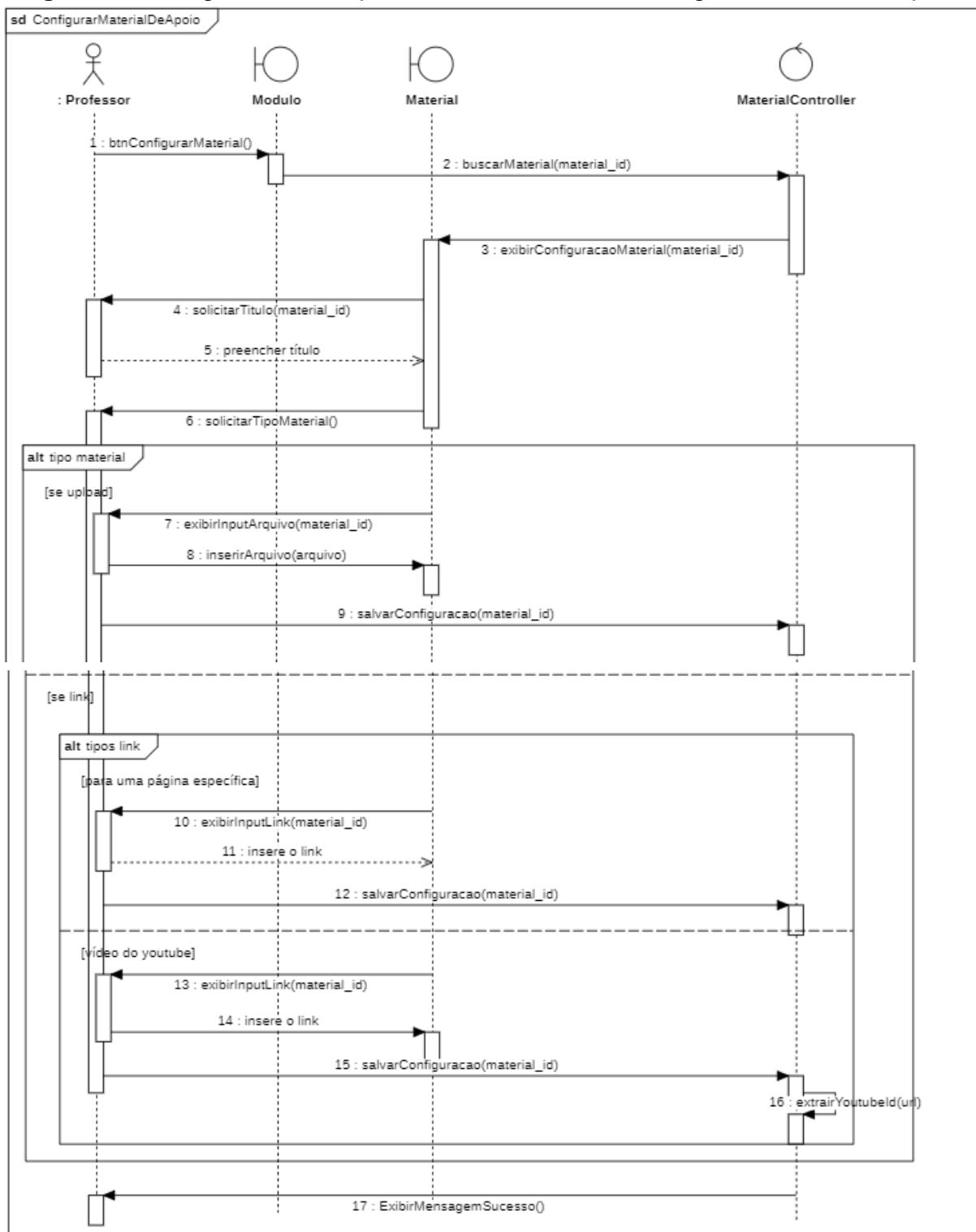
Fonte: Elaborado pelo próprio autor.

**Figura 17 – Diagrama de Sequência do caso de uso Configurar pergunta.**



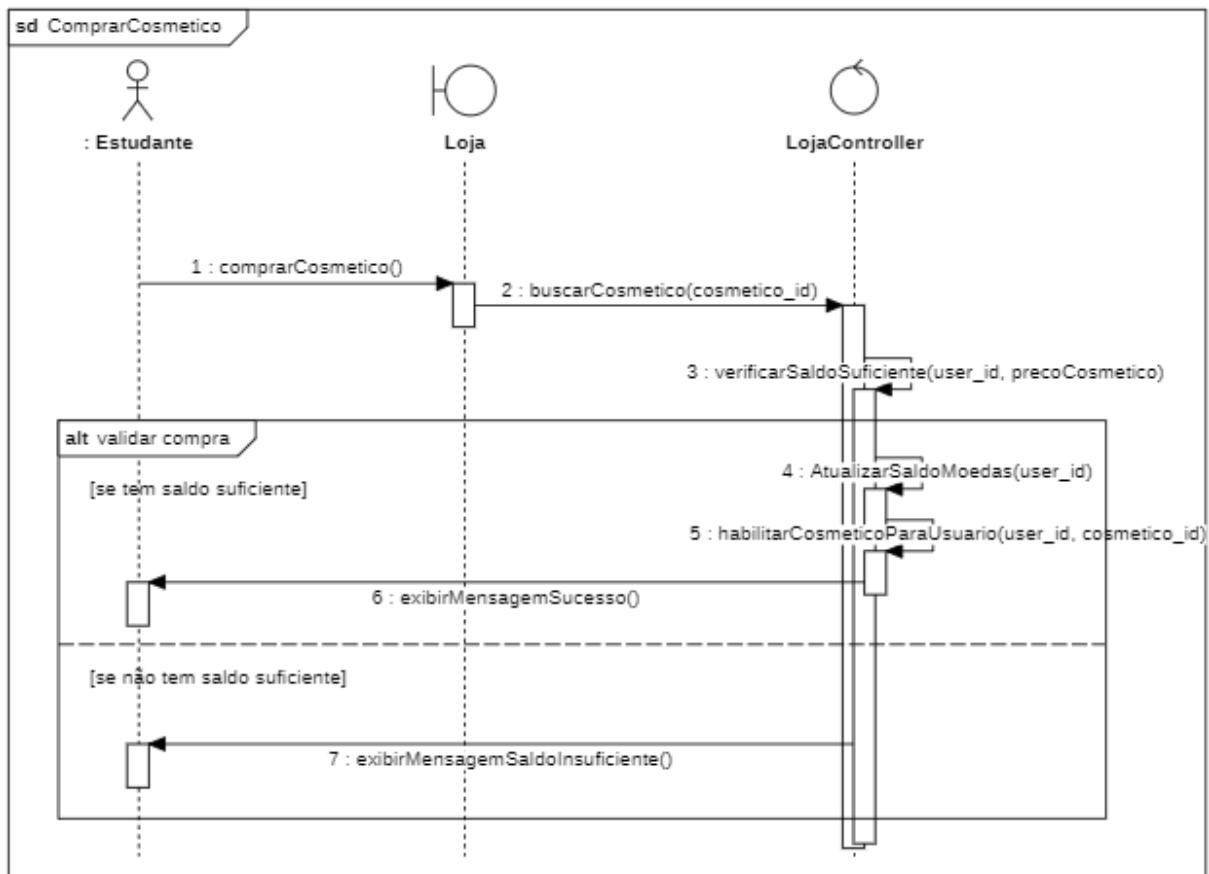
Fonte: Elaborado pelo próprio autor.

**Figura 18 – Diagrama de Sequência do caso de uso Configurar material de apoio.**



Fonte: Elaborado pelo próprio autor.

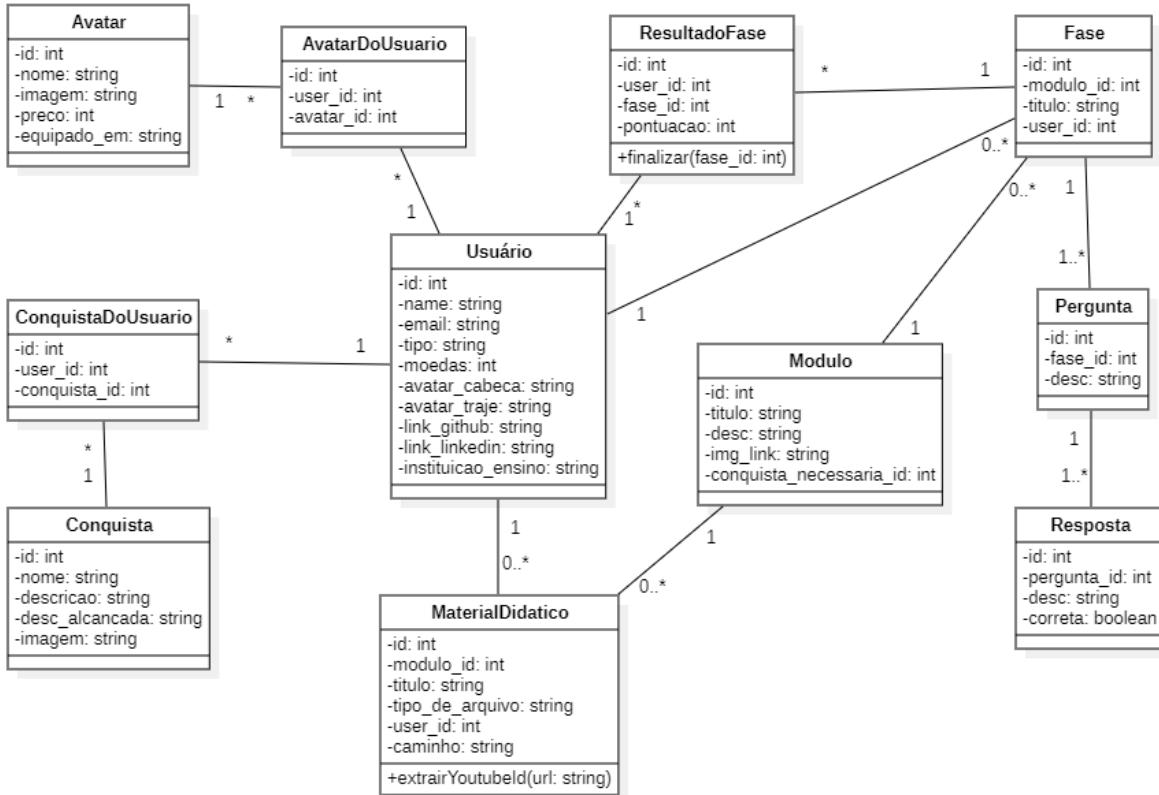
**Figura 19 – Diagrama de Sequência do caso de uso Comprar cosmético.**



Fonte: Elaborado pelo próprio autor.

## 4.2 DIAGRAMA DE CLASSES

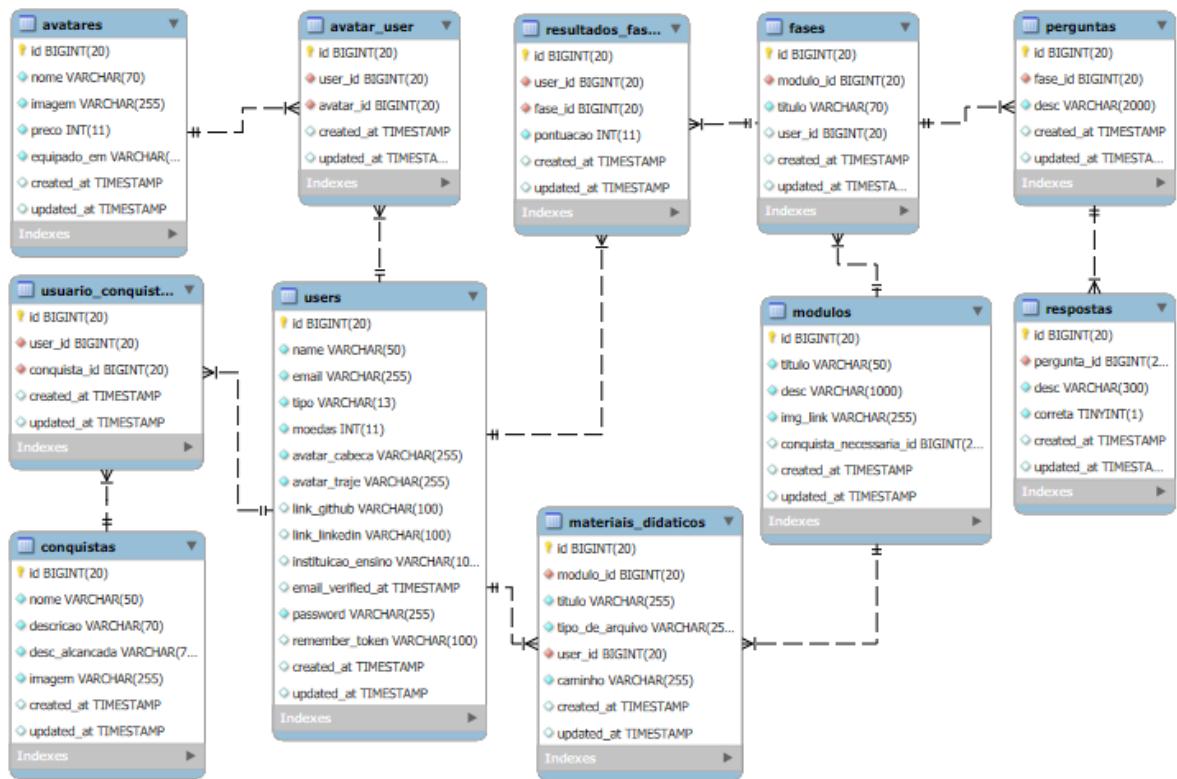
**Figura 20** – Diagrama de Classes.



Fonte: Elaborado pelo próprio autor.

### 4.3 MODELAGEM DA BASE DE DADOS

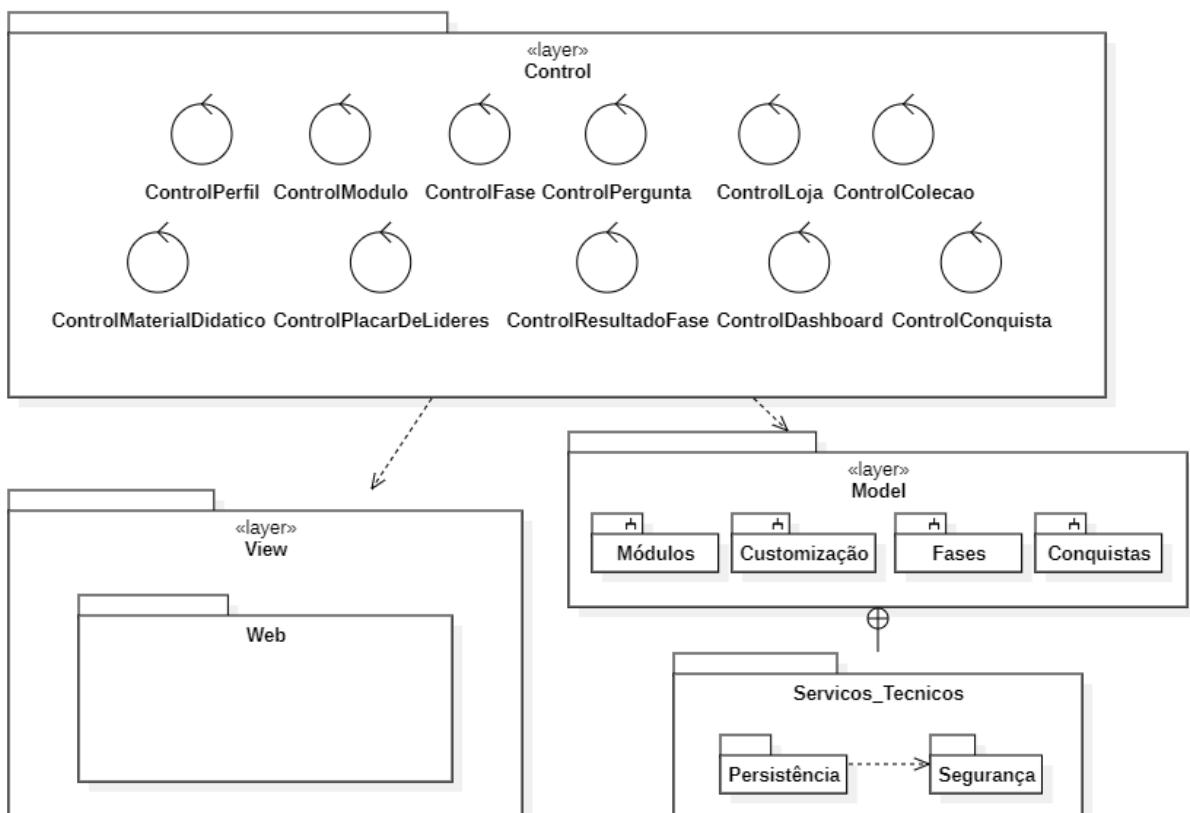
**Figura 21** – Modelagem da Base de Dados.



Fonte: Elaborado pelo próprio autor.

#### 4.4 DIAGRAMA DE PACOTES DA ARQUITETURA LÓGICA

**Figura 22** – Diagrama de Pacotes da Arquitetura Lógica.

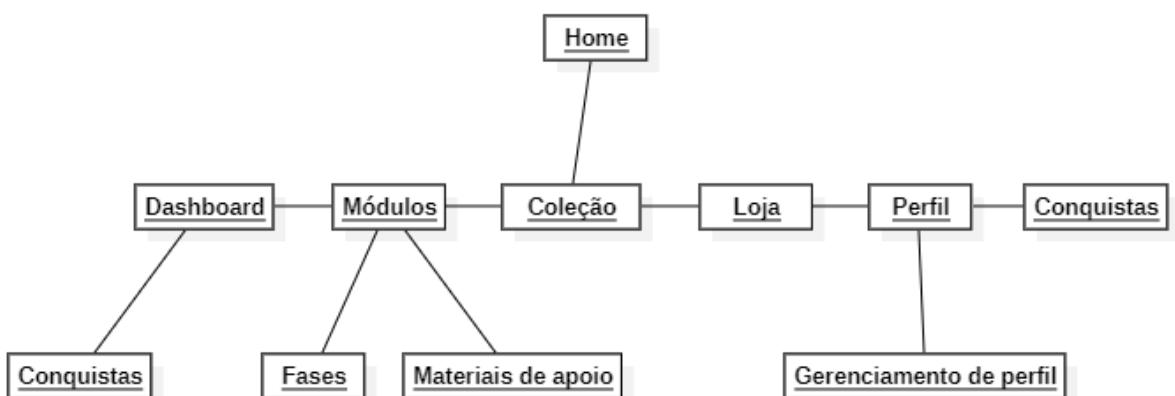


Fonte: Elaborado pelo próprio autor.

#### 4.5 MODELO NAVEGACIONAL

Para usuários do tipo aluno:

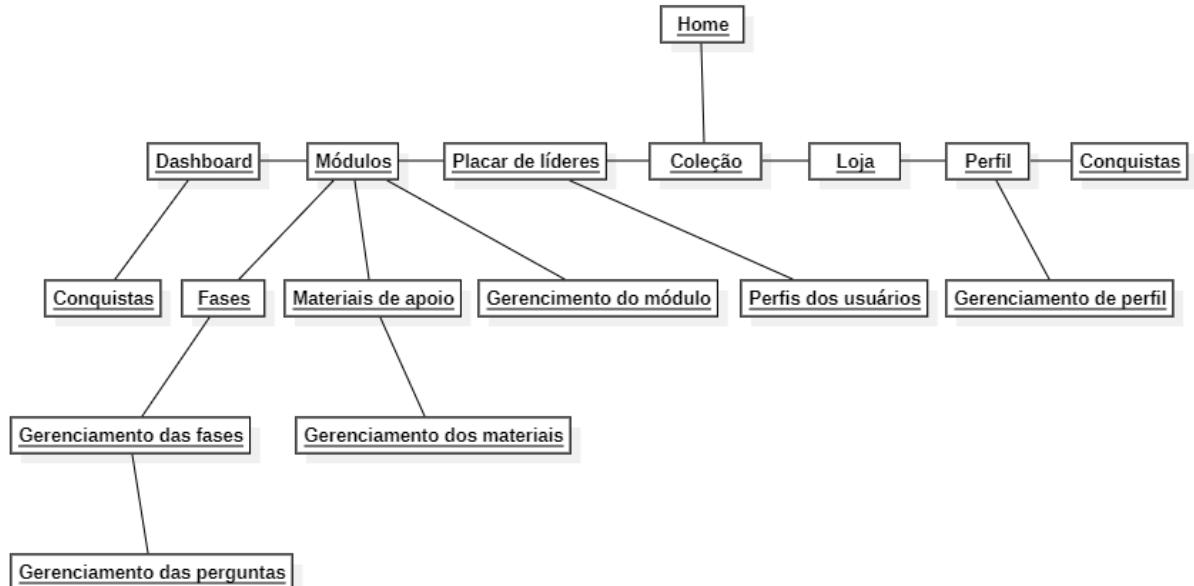
**Figura 23** – Modelo Navegacional para usuários do tipo aluno.



Fonte: Elaborado pelo próprio autor.

Para usuários do tipo professor:

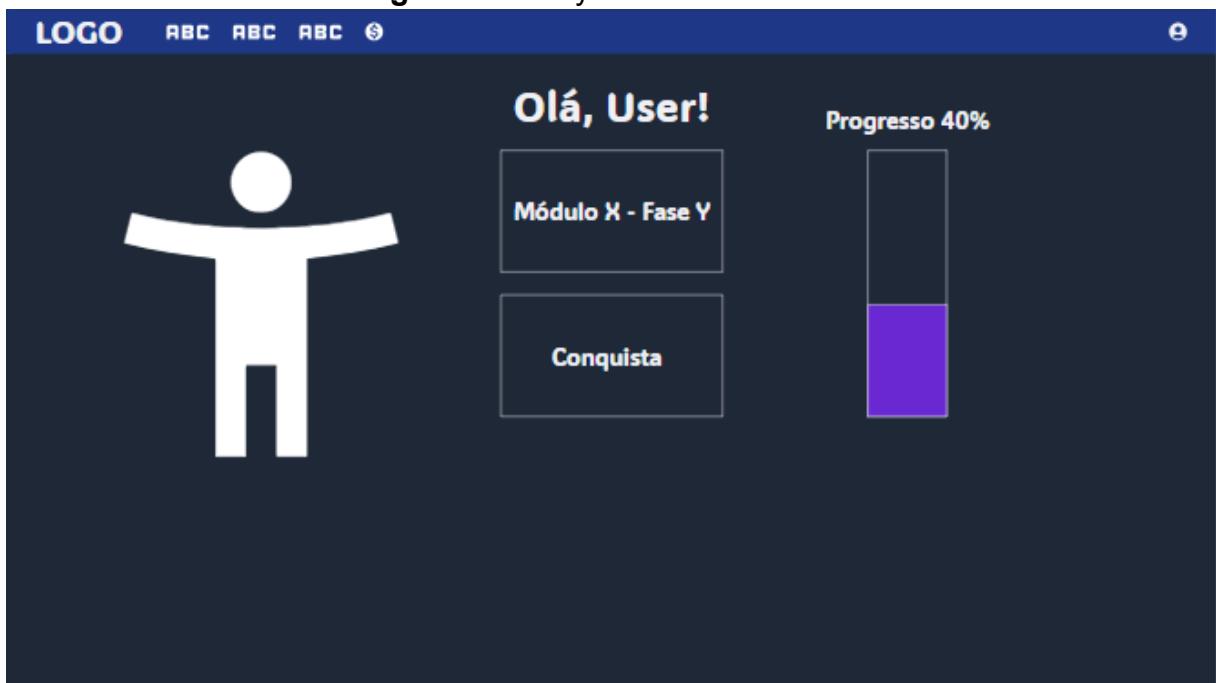
**Figura 24** – Modelo Navegacional para usuários do tipo professor.



Fonte: Elaborado pelo próprio autor.

#### 4.6 OUTROS LAYOUTS DE TELAS

**Figura 25** – Layout do Dashboard.



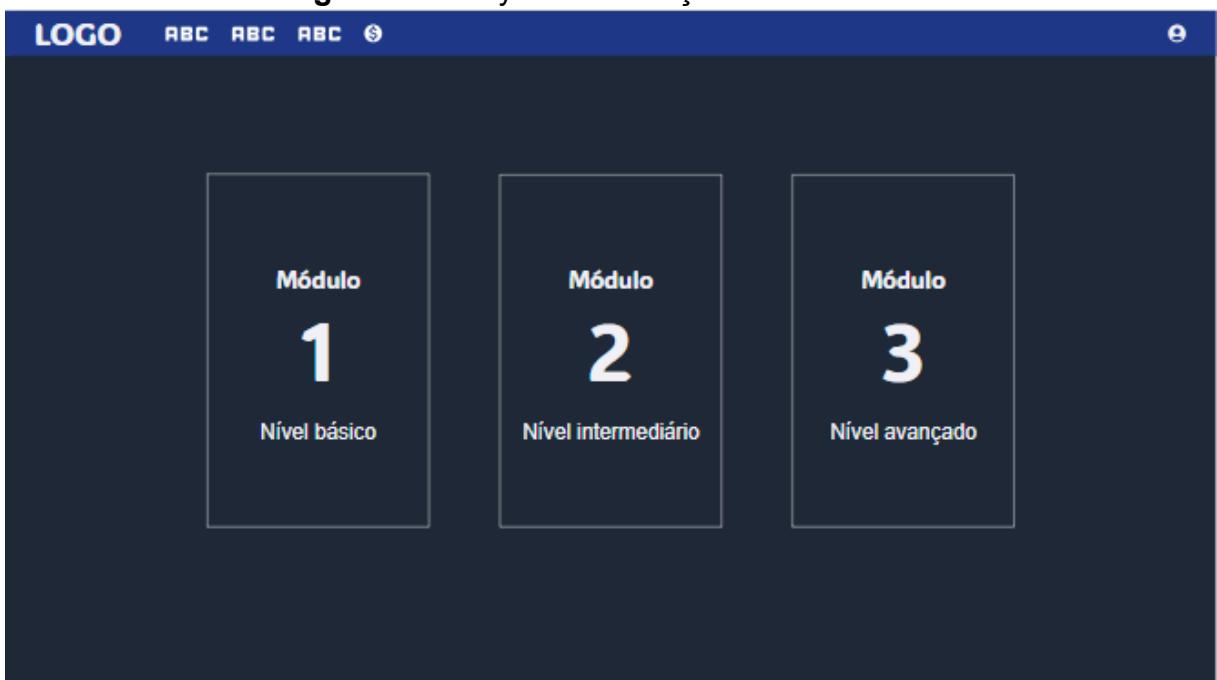
Fonte: Elaborado pelo próprio autor.

**Figura 26 – Layout do placar de líderes.**



Fonte: Elaborado pelo próprio autor.

**Figura 27 – Layout da exibição dos módulos.**



Fonte: Elaborado pelo próprio autor.

Figura 28 – Layout do conteúdo dos módulos.



The screenshot shows a mobile application interface. At the top, there is a blue header bar with the text "LOGO" and "ABC ABC ABC" on the left, and a user icon on the right. Below the header, the title "Módulo X" is centered. The main content area is divided into two sections: "Conteúdo base" and "Fases".

**Conteúdo base:**

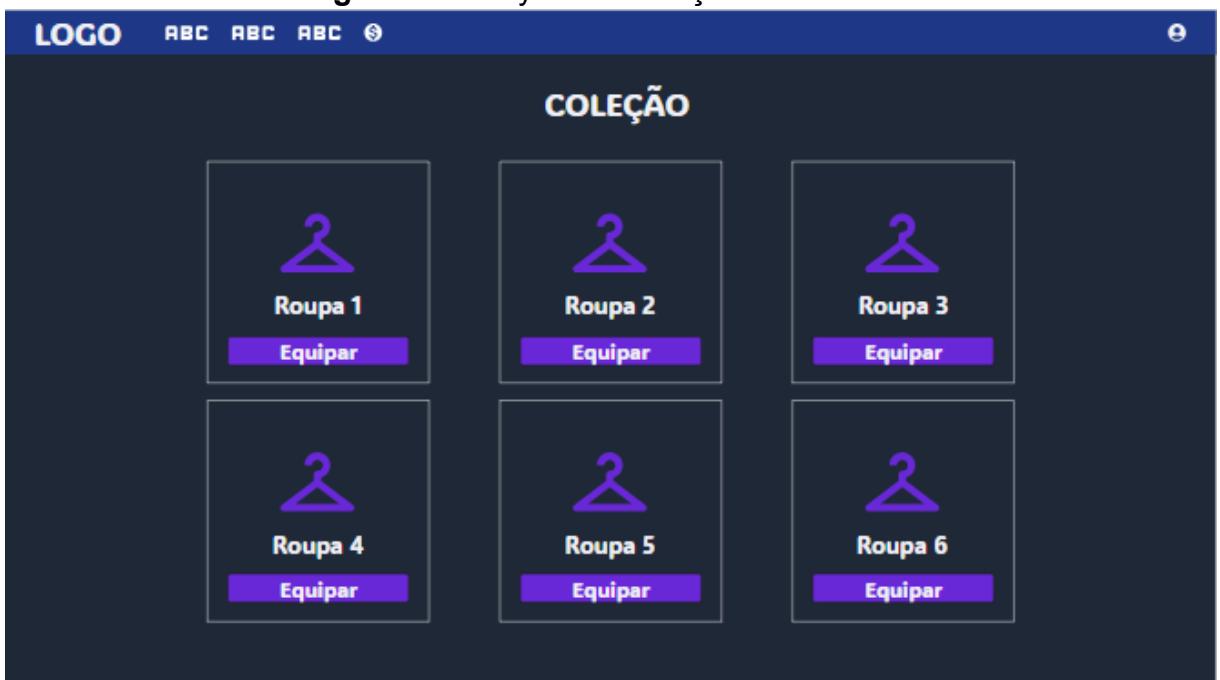
- Descrição: Material didático - A crise do software.pdf
- Descrição: Video - Introdução à Engenharia de Software
- ... (ellipsis)

**Fases:**

- Fase 1 - Sobre os mitos
- Fase 2 - A crise do software
- Fase 3 - Identificando os modelos de processos de software
- ... (ellipsis)

Fonte: Elaborado pelo próprio autor.

Figura 29 – Layout da coleção do usuário.



The screenshot shows a mobile application interface. At the top, there is a blue header bar with the text "LOGO" and "ABC ABC ABC" on the left, and a user icon on the right. Below the header, the title "COLEÇÃO" is centered. The main content area displays six items, each represented by a purple hanger icon and labeled "Roupa 1" through "Roupa 6". Each item has a purple "Equipar" button at the bottom.

Roupa	Equipar
Roupa 1	Equipar
Roupa 2	Equipar
Roupa 3	Equipar
Roupa 4	Equipar
Roupa 5	Equipar
Roupa 6	Equipar

Fonte: Elaborado pelo próprio autor.

**Figura 30** – Layout das conquistas do usuário.



Fonte: Elaborado pelo próprio autor.

**Figura 31** – Layout da exibição do perfil de um usuário.



Fonte: Elaborado pelo próprio autor.

## APÊNDICE A – Procedimentos para Implantação do Sistema

Para a implantação do sistema ProES, é necessário configurar um ambiente que ofereça suporte às tecnologias utilizadas no desenvolvimento da aplicação. O sistema foi desenvolvido com o framework Laravel, utilizando o Blade como motor de templates e o Laravel Breeze como starter kit para autenticação e estruturação inicial. Embora o banco de dados escolhido tenha sido MySQL, um sistema de gerenciamento relacional amplamente utilizado, a configuração pode ser feita de acordo com a necessidade, ou vontade, já que o laravel permite essa flexibilidade.

O servidor web pode ser configurado com Apache. É imprescindível que o PHP esteja instalado na versão 8.0 ou superior, juntamente com extensões necessárias como OpenSSL, PDO, Mbstring, Tokenizer, XML, Ctype, JSON. Além disso, é necessário dispor do Composer para gerenciamento das dependências PHP e do Node.js, juntamente com o NPM ou Yarn, para a compilação dos arquivos front-end.

O processo de implantação inicia-se com a clonagem do repositório da aplicação no servidor de hospedagem. Em seguida, é realizada a instalação das dependências PHP através do comando “composer install”. Após essa etapa, deve-se configurar o arquivo de variáveis de ambiente, copiando o “.env.example” para “.env” e ajustando as configurações necessárias, como as credenciais de acesso ao banco de dados e a chave de aplicação, que deve ser gerada com o comando “php artisan key:generate”. O banco de dados MySQL precisa ser criado previamente, e suas credenciais devem ser configuradas corretamente no arquivo “.env”.

Com o banco de dados configurado, executam-se as migrações por meio do comando “php artisan migrate”, que cria as tabelas e estrutura necessárias para o funcionamento da aplicação. Na sequência, as dependências JavaScript são instaladas utilizando “npm install”, e os arquivos front-end são compilados com “npm run dev” para ambientes de desenvolvimento ou “npm run build” para produção. Também é fundamental garantir que as permissões das pastas “storage” e “bootstrap/cache” estejam adequadas para leitura e escrita pelo servidor web.

Por fim, deve-se configurar o servidor web para apontar para o diretório “public” da aplicação, garantindo que todas as rotas sejam corretamente direcionadas pelo Laravel. Caso a implantação seja realizada em ambientes de nuvem, como DigitalOcean, AWS ou Heroku, recomenda-se a configuração de certificados SSL, políticas de segurança, como firewalls, e rotinas de backup automático do banco de dados, além de monitoramento contínuo de recursos para assegurar a estabilidade e a escalabilidade do sistema.

## ANEXO 1 – Referências

- ENADE (2008). Prova de Tecnologia em Análise e Desenvolvimento de Sistemas. Inep, [Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enade/provas-e-gabaritos>].
- ENADE (2011). Tecnologia em Análise e Desenvolvimento de Sistemas. Inep, [Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enade/provas-e-gabaritos>].
- ENADE (2014). Tecnologia em Análise e Desenvolvimento de Sistemas. Inep, [Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enade/provas-e-gabaritos>].
- ENADE (2017). Tecnologia em Análise e Desenvolvimento de Sistemas. Inep, [Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enade/provas-e-gabaritos>].
- ENADE (2021). Tecnologia em Análise e Desenvolvimento de Sistemas. Inep, [Disponível em: <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enade/provas-e-gabaritos>].

**ANEXO 2 – Manual do Usuário**

1. INTRODUÇÃO.....	1
2. ACESSO AO SISTEMA.....	1
3. CREDENCIAMENTO.....	1
4. PRIMEIROS PASSOS.....	1
5. NAVEGAÇÃO.....	2
6. APRENDIZADO E DESAFIOS.....	2
7. RECOMPENSAS.....	4
8. CONQUISTAS.....	4
9. IDENTIFIQUE-SE.....	5
11. SUPORTE.....	8

## 1. INTRODUÇÃO

Bem-vindo ao ProES, uma plataforma gamificada desenvolvida para auxiliar no ensino e aprendizado da disciplina de Engenharia de Software.

Este manual tem como objetivo orientá-lo no uso das principais funcionalidades do sistema, garantindo uma experiência completa e eficiente.

## 2. ACESSO AO SISTEMA

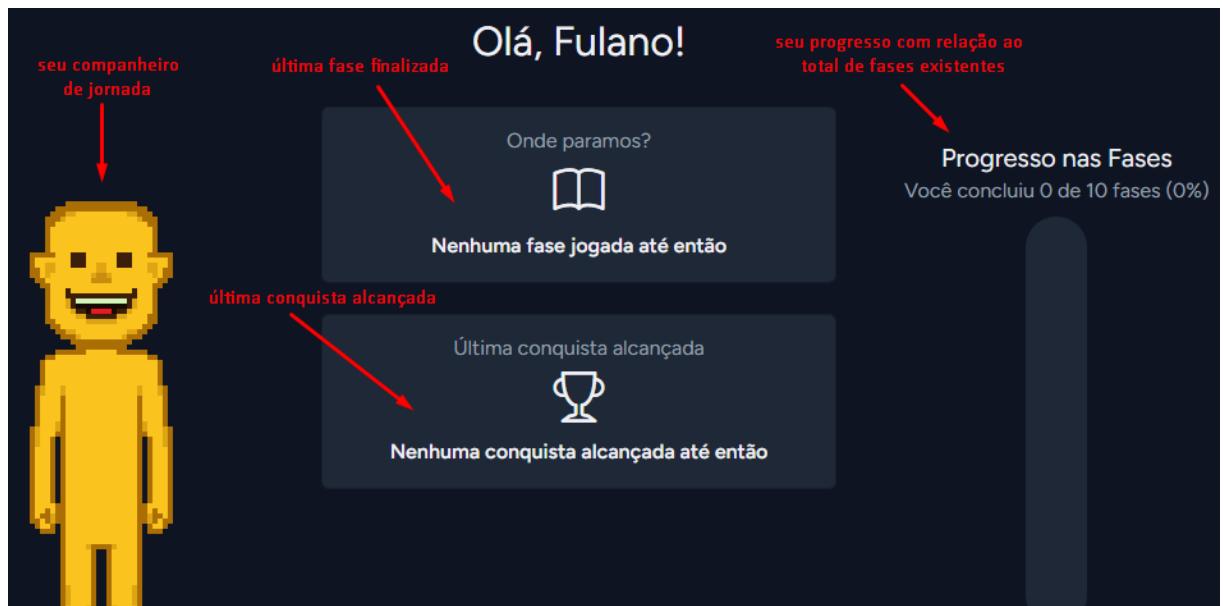
Para o acessar o sistema será necessário fazer o uso de um navegador atualizado e, consequentemente, conexão com a internet.

## 3. CREDENCIAMENTO

Para ter acesso à plataforma será necessário ter uma conta e para isso, ao acessar o site você pode clicar em “Registrar”, para o caso onde você já tenha uma conta, basta acessá-la pela opção “Entrar”.

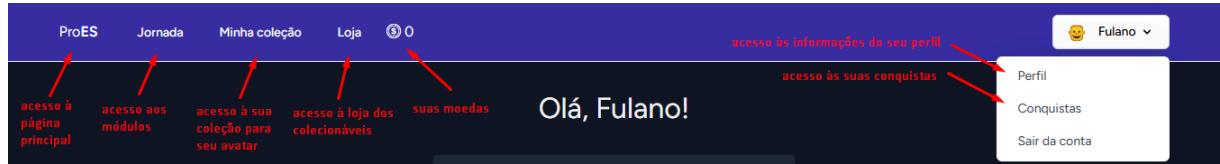
## 4. PRIMEIROS PASSOS

Ao acessar você logo se depara com um menu principal onde você pode visualizar seus últimos feitos, seu progresso e o personagem que lhe acompanha durante sua jornada, veja:



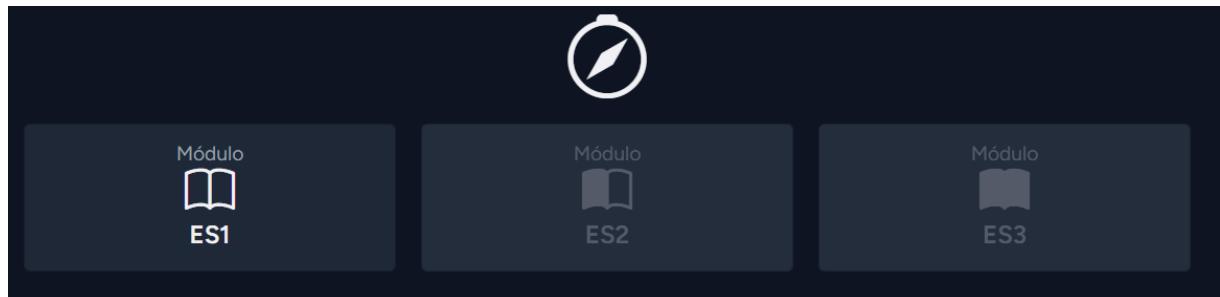
## 5. NAVEGAÇÃO

A navegação pelo site pode ser feita, na maioria das situações, pela própria barra superior.



## 6. APRENDIZADO E DESAFIOS

Ao acessar os módulos você irá notar que existe um bloqueio do acesso aos módulos seguintes, justamente para que você faça o percurso intuitivo e aprenda o necessário para continuar.



A estrutura do módulo é composta pelos materiais de apoio e pelas fases, tais materiais podem estar disponibilizados em alguns formatos sendo eles: link externo, vídeo do youtube, de extensão mp4, imagem ou algum arquivo compactado, de todos eles o único que será baixado automaticamente será o arquivo zip, os demais serão abertos em uma nova guia, aproveite-os!

Este módulo contempla conteúdos como crise do software, modelos de processo de software entre outros introdutórios para a disciplina.

Material de apoio

Crise do software

Vamos jogar!

Modelos de Processo de Software

Definições de requisitos

Já quanto às fases, quando você às acessa poderá finalizá-la no tempo que precisar, nelas estão contidas quizzes como este:

- R1: o sistema deve permitir que cada professor realize o lançamento de notas das turmas nas quais lecionou;
- R2: o sistema deverá ser desenvolvido de forma a possibilitar seu transporte para outro sistema operacional em, no máximo, sessenta dias;
- R3: o sistema deve permitir que um estudante realize a sua matrícula nas disciplinas oferecidas em um semestre letivo;
- R4: o sistema atualiza a nota do estudante, permitindo sua visualização, em até dois segundos depois do momento que o professor a registra;
- R5: o sistema deve permitir que o auxiliar de serviços acadêmicos realize o cadastro de um estudante em não mais do que dez minutos de orientação.

- R1, R2 e R3.
- R1, R2 e R5.
- R1, R3 e R4.
- R2, R4 e R5.
- R3, R4 e R5.

[Anterior](#)

[Enviar respostas](#)

Ao clicar em “Enviar respostas” você finaliza sua tentativa e recebe suas recompensas por ela, daí vai uma informação, toda fase tem moedas finitas, uma para cada resposta acertada, ao conseguir responder corretamente às perguntas da fase você passa a não receber mais moedas, então use-as com sabedoria!

Você ganhou 1 moeda!

## 7. RECOMPENSAS

As recompensas podem ser usadas para deixar seu companheiro mais à sua cara, visite a loja e faça umas compras... Mas cuidado, se seu saldo não for suficiente não vai ser possível comprar o acessório que deseja.



Com o cosmético comprado você pode ir até a sua coleção e equipá-lo, após fazê-lo, volte à página principal e veja a diferença!



## 8. CONQUISTAS

As recompensas não param por aí, explore os conteúdos e avance na plataforma para ter algumas surpresas...



## 9. IDENTIFIQUE-SE

Você não precisa, mas se quiser, identifique-se na plataforma incrementando seu perfil...



## 10. FUNCIONALIDADES EXCLUSIVAS DO PROFESSOR

Algumas funções somente poderão ser desenvolvidas pelos professores, são elas: o gerenciamento dos módulos, dos materiais e das fases. E isso poderá ser feito justamente quando entrar num módulo.

Note que existem algumas opções a mais que um estudante, vale ressaltar que tais opções só aparecerão para o professor que criou o material/fase, essas opções têm suas versões gerais e individuais, então se você quiser adicionar uma nova pergunta em uma fase já existente (que seja sua) basta acessar a edição da fase, caso queria adicionar uma nova fase de fato, basta iniciá-la do zero com o botão do sinal de adição. Muito cuidado com os botões gerais, o de exclusão, por exemplo, além de excluir a fase, também exclui as perguntas e respostas à ela vinculadas.

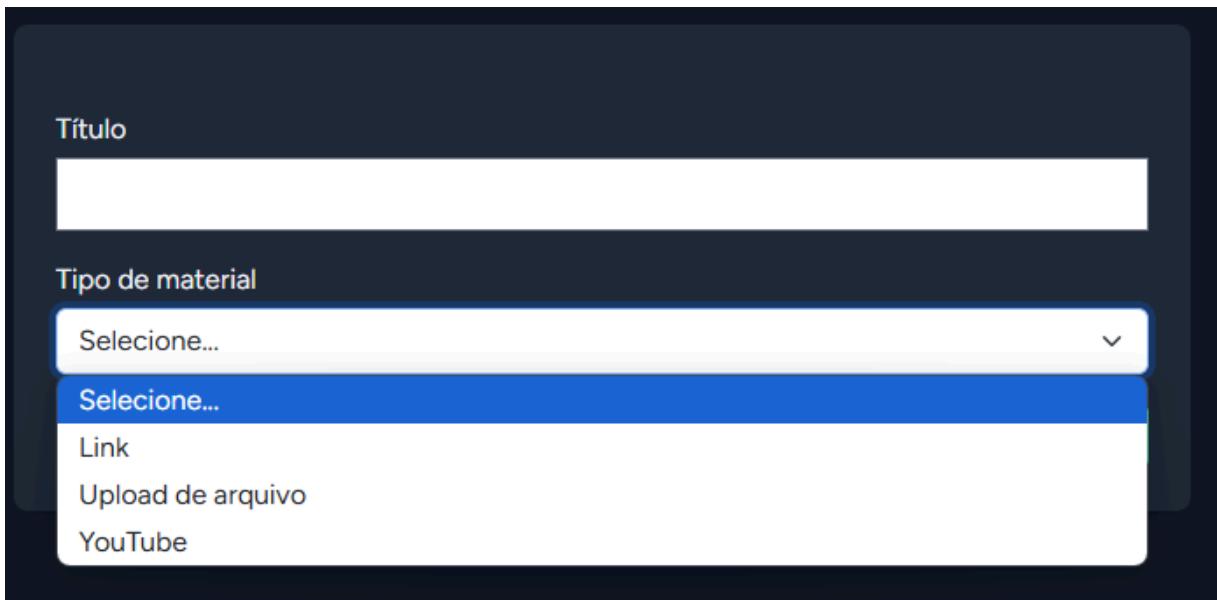
Para deixar claro então, os seguintes botões permitem adicionar um material e uma fase, respectivamente:

Já botões como os seguintes permitem editar e excluir o que referenciam:

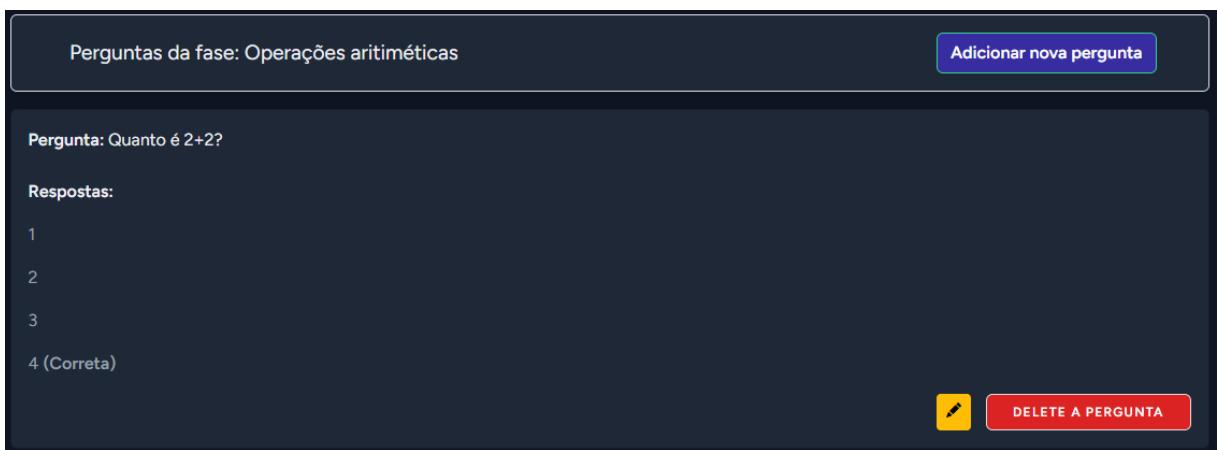


Os materiais de apoio aceitam vários tipos de arquivos, todavia atente-se aos requisitos mínimos do tipo de material, se for um link é importante que seja um link válido, senão os estudantes serão levados para lugar nenhum. Se a opção for

arquivo, não esqueça de selecioná-lo. E por fim, se for um link do Youtube coloque-o por inteiro que a plataforma saberá o que fazer!



Se notar algum erro nas fases que você fez, poderá por meio da opção de edição alterar uma pergunta individualmente, seja o erro na pergunta, nas respostas ou na seleção de qual delas é a correta, a edição só não lhe permitirá colocar novas respostas na pergunta, para tal exclua a pergunta e a refaça na opção “Adicionar nova pergunta” da página de gerenciamento das mesmas.



O módulo contém uma descrição que pode ser alterada, dependendo da dinâmica da disciplina é até possível colocar a ementa da mesma, tem bastante espaço para tal, ele será exibido como foi escrito.

ES1

Descreva o módulo (insira informações como os objetivos, habilidades e competências adquiridas pelos alunos, como na ementa)

Isso é uma ementa exemplo, nela descreverei algo genérico.

Objetivos da disciplina:

- \* Objetivo genérico 1
- \* Objetivo genérico 2
- \* Objetivo genérico 3

**Cancelar** **Salvar alterações**

E por fim, mas não menos importante o placar de líderes, por ele é possível acessar o perfil dos alunos e visualizar seus contatos, se tiverem incrementado previamente.

Placar de líderes de pontuação	
Professor	6 pontos
Fulano	1 ponto

## 11. SUPORTE

A possibilidade de reportar algum defeito “bug”, contribuição ou alguma sugestão para implementação, efetue diretamente no repositório do projeto no GitHub via link: <https://github.com/gabriel-smantovani/proes>