**GREEM**

**Portal do Aluno para reforços escolares — Especificação de Requisitos de Software Para Divulgação e procura de aulas de reforço escolar.**

**Versão 1.7**

**UNINASSAU — Sistemas de informação, 8º período**

**Maria Elysa da S Barbosa**

**Gabriel Matheus**

**Marcone Alexandre**

**Rubenilson Lira**

**Elisson Guimel**

**Histórico da Revisão**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| **01/09/2020** | **1.0** | **Criação do documento** | **Maria Elysa da S Barbosa** |
| **02/09/2020** | **1.1** | **Inserção das sessões de requisitos** | **Maria Elysa da S Barbosa** |
| **03/09/2020** | **1.2** | **Inserção das sessões “Informações de suporte” e “Restrições de design”** | **Maria Elysa da S Barbosa** |
| **03/09/2020** | **1.3** | **Inserção da sessão “Interfaces” e “Documentação do usuário on-line e requisitos do Sistema de Ajuda” e “Padrões Aplicáveis”** | **Gabriel Matheus De Lemos Soares** |
| **09/09/2020** | **1.4** | **Atualização do documento** | **Maria Elysa da S Barbosa** |
| **23/09/2020** | **1.5** | **Atualização dos requisitos funcionais** | **Maria Elysa da S Barbosa** |
| **01/10/2020** | **1.6** | **Inserção da sessão Diferenciais e Alternativas e concorrências** | **Maria Elysa da Silva Barbosa** |
| **06/11/2020** | **1.7** | **Atualização da sessão de requisitos funcionais** | **Maria Elysa da Silva Barbosa** |

**Índice**

1. Introdução

1.1 Objetivo

1.2 Escopo

1.3 Definições, Acrônimos e Abreviações

1.4 Referências

1.5 Visão Geral

2. Descrição Geral

3. Requisitos Específicos

3.1 Funcionalidade

3.1.1 Cadastrar professor

3.1.2 Cadastrar aluno

3.1.3 Editar aluno

3.1.4 Excluir aluno

3.1.5 Anexar arquivos

3.1.6 Excluir arquivos anexados

3.1.7 Registrar desempenho do aluno

3.1.8 Login do professor

3.1.9 Login do aluno

3.1.10 Login do pai/responsável

3.1.11 Anunciar professor (Aulas de reforço)

3.1.12 Buscar professor

3.2 Utilidade

3.2.1Permitir entrada via teclado e mouse

3.2.2 Elementos padronizados

3.3 Confiabilidade

3.3.1 Disponibilidade

3.3.2 Política de criação de senhas

3.3.3 Sigilo das informações

3.3.4 Integridade dos dados

3.3.5 Política de criação de contas de usuários

3.4 Desempenho

3.4.1 Tempo de resposta

3.4.2 Capacidade

3.4.3 Rede

3.5 Suportabilidade

3.5.1 Favorecer a orientação do usuário.

3.5.2 Campos intuitivos e autoexplicativos

3.5.3 Adequação aos navegadores

3.5.4 Facilidade para recuperação de falhas

3.5.5 Suporte ao usuário

3.6 Restrições de Design

3.6.1 Restrição de modelo de arquitetura

3.6.2 Restrição de linguagens e desenvolvimento

3.6.3Restrição de banco de dados

3.7 Documentação do Usuário On-line e Requisitos do Sistema de Ajuda

3.8 Componentes Comprados

3.9 Interfaces

3.9.1 Interfaces com o Usuário

3.9.2 Interfaces de Hardware

3.9.3 Interfaces de Software

3.9.4 Interfaces de Comunicações

3.10 Requisitos de Licença

3.11 Observações Legais, sobre Direitos Autorais e Outras Observações

3.12 Padrões Aplicáveis

4. Informações de Suporte

5. Alternativas e concorrências

6. Diferenciais

**Especificação de Requisitos de Software**

**1. Introdução**

Neste documento está, de forma compacta e de fácil entendimento, todo o levantamento de requisitos do sistema aqui apresentado, sendo os requisitos funcionais e não funcionais. Também são apresentados os casos de uso e User Stories criados a partir dos documentos levantados pela equipe juntamente com o acompanhamento do orientador do projeto.

**1.1 Objetivo**

Este documento tem como objetivo registrar e especificar os requisitos que devem ser atendidos pelo Portal do Aluno para Reforços Escolares, de forma a satisfazer as necessidades de seus clientes e auxiliar os desenvolvedores com relação as funcionalidades esperadas pelo produto.

**1.2 Escopo**

* Identificar pelo nome o produto de software a ser produzido (Portal do Aluno para Reforços Escolares)
* Conduzir uma visão geral do documento, sinalizando as funções, condições e restrições do produto.
* Esclarecer de modo breve quem são os usuários (atores).
* Descrever os requisitos funcionais e não funcionais do produto.

**1.3 Definições, Acrônimos e Abreviações**

MVC:é o acrônimo de Model-View-Controller

Atores:São os proprietários da história, no caso, quem irá utilizar ou está interessado na funcionalidade ou requisito.

RF: Requisito funcional

RNF: Requisito não funcional

**1.4 Referências**

* Documento de Requisitos

**1.5 Visão Geral**

Este documento define de forma organizada uma descrição geral do sistema proposto, e logo em seguida descreve suas funcionalidades especificando as entradas e saídas para todos os requisitos funcionais. Faz também uma descrição clara dos requisitos não funcionais contidos neste sistema.

**2. Descrição Geral**

O Portal do Aluno para Reforços Escolares gerencia de forma prática o desenvolvimento acadêmico dos alunos. De modo que o professor possa registrar, editar e excluir dados como notas de avaliações, materiais didáticos, frequência e horário de aulas. O sistema também permite uma tela diferente para os alunos e pais ou responsáveis, onde eles podem visualizar suas informações — ou a de deus filhos — registradas previamente pelo professor do reforço escolar. E mais um diferencial do sistema é que pais e alunos podem achar professores por sua redondeza com a ajuda dos “anúncios”, criados pelos professores como meio de divulgar suas aulas.

**3. Requisitos Específicos**

**3.1 Funcionalidade**

3.1.1 Cadastrar professor

|  |  |
| --- | --- |
| **RF01** | |
| NOME | Cadastrar professor |
| DESCRIÇÃO | O sistema deve permitir que o professor se cadastre na plataforma. Usando informações básicas como, por exemplo, e-mail, cpf e celular. |
| ATORES | Professor |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Professor seleciona os campos e registra nome completo, endereço, cpf, telefone e e-mail |
| SAÍDAS E PÓS-CONDIÇÕES | Professor conclui cadastro com sucesso |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Professor abre o aplicativo/site |
| 2.     A tela inicial aparece |
| 3.     Clica no botão “Cadastre-se” |
| 4.     Insere os dados nos campos correspondentes a Nome, cpf, email, endereço, telefone, aulas que leciona. |
| 5.     Todos os campos são validados, cadastro concluído com sucesso. |
| FLUXO SECUNDÁRIO: | 5. Algum dos campos não é validado com sucesso e o cadastro não é concluído com sucesso. |
| 6. Usuário retorna para os campos e revisa os dados até que todos sejam validados com sucesso. |

3.1.2 Cadastrar aluno

|  |  |
| --- | --- |
| **RF02** | |
| NOME | Cadastrar aluno |
| DESCRIÇÃO | O sistema deve permitir que o professor, depois de previamente cadastrado na plataforma, cadastre seus alunos de forma individual, assim, ligando automaticamente eles ao professor que o cadastrou. |
| ATORES | Professor |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Professor seleciona os campos e registra nome completo do aluno, endereço, telefone dele e/ou do responsável. |
| SAÍDAS E PÓS-CONDIÇÕES | Professor conclui cadastro com sucesso |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Professor abre o aplicativo/site |
| 2.     Clica no botão de “login” |
| 3.     Entra com o e-mail e senha |
| 4.     Clica na aba de cadastrar aluno |
| 5.     Insere nos campos correspondentes os dados do aluno, como nome, nome dos responsáveis, e-mail dos responsáveis, endereço, telefone, telefone dos responsáveis, data de nascimento. |
|  | 6. Dados são válidos e aluno é cadastrado. |
| FLUXO SECUNDÁRIO: | 6. Dados não são validados e o aluno não é cadastrado. |
| 7. Professor retorna aos dados informados e revisa até todos estarem corretos e sem validados, concluindo o cadastro. |

3.1.3 Editar aluno

O sistema deve permitir que o professor edite o registro de aluno, podendo assim, fazer correções.

3.1.4 Excluir aluno

O sistema deve permitir que o professor exclua o registro de um aluno que ele mesmo já tenha cadastrado. Podendo assim, retirá-lo de sua lista de matriculados quando o aluno concluir suas aulas particulares.

3.1.5 Anexar arquivos

|  |  |
| --- | --- |
| **RF05** | |
| NOME | Anexar arquivos |
| DESCRIÇÃO | O sistema deve permitir que o professor, depois de cadastrado, possa anexar arquivos (pdf, .doc, links, entre outros), que podem ser utilizados como complemento para algum assunto proposto em classe. |
| ATORES | Professor |
| PRIORIDADE | Média |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Professor seleciona o campo e anexa o material. |
| SAÍDAS E PÓS-CONDIÇÕES | Professor conclui anexo com sucesso. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Professor abre app/site |
| 2.     Clica no botão de “login” |
| 3.     Entra com seu e-mail e senha |
| 4.     Clica na aba de alunos cadastrados |
| 5.     Escolhe o aluno |
| 6.     Clica na aba de anexos |
| 7.     Adiciona o anexo com sucesso |
| FLUXO SECUNDÁRIO: | 7. Anexo não adicionado com sucesso |
| 8.     Professor revisa formato de arquivo adicionado. |

3.1.6 Excluir arquivos anexados

O sistema deve permitir que o professor, depois de cadastrado, possa excluir arquivos que ele já tenha anexado na plataforma. Assim, podendo sempre atualizar seus materiais de ensino.

3.1.7 Registrar desempenho do aluno

|  |  |
| --- | --- |
| **RF07** | |
| NOME | Registrar Desempenho do aluno |
| DESCRIÇÃO | O sistema deve permitir que o professor, depois de cadastrado, possa registrar o desempenho do aluno. Seja semanalmente/mensalmente/anualmente. Assim, podendo acompanhar o crescimento do aluno de modo mais abrangente. |
| ATORES | Professor |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Professor seleciona o campo e registra o desempenho do aluno, seja por uma nota descritiva ou utilizando de valores numéricos. |
| SAÍDAS E PÓS-CONDIÇÕES | Professor registra o desempenho do aluno com sucesso. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Professor abre app/site |
| 2.     Clica no botão de “login” |
| 3.     Entra com seu e-mail e senha |
| 4.     Clica na aba de alunos cadastrados |
| 5.     Escolhe o aluno |
| 6.     Clica na aba de desempenho |
| 7.     Registra o desempenho do aluno com sucesso |
| FLUXO SECUNDÁRIO: | 7.Não registra desempenho do aluno com sucesso. |
| 8. Professor retorna aos campos de registro de desempenho |
| 9. Revisa os dados informados. |
| 10. Conclui o registro com sucesso |

3.1.8 Login do professor

|  |  |
| --- | --- |
| **RF08** | |
| NOME | Login do professor |
| DESCRIÇÃO | O sistema deve permitir que o professor, depois de cadastrado, possa fazer seu login no sistema utilizando apenas o nome de login e senha. |
| ATORES | Professor |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Professor seleciona os campos e adiciona o login e a senha. |
| SAÍDAS E PÓS-CONDIÇÕES | Professor consegue logar no sistema com sucesso. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Professor abre app/site |
| 2.     Clica no botão de “login” |
| 3.     Entra com e-mail e senha criados no ato do cadastro. |
| 4.     Professor loga no sistema com sucesso. |
| FLUXO SECUNDÁRIO | 4. Professor não consegue logar no sistema com sucesso. |
| 5. Retorna aos campos de e-mail e senha |
| 6. Revisa os dados informados. |

3.1.9 Login do aluno

|  |  |
| --- | --- |
| **RF09** | |
| NOME | Login do aluno |
| DESCRIÇÃO | O sistema deve permitir que o aluno, depois de cadastrado pelo professor, possa fazer seu login no sistema utilizando apenas o nome de login ou e-mail e senha. |
| ATORES | Aluno |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | aluno seleciona os campos e adiciona o login e a senha. |
| SAÍDAS E PÓS-CONDIÇÕES | aluno consegue logar no sistema com sucesso. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Aluno abre app/site |
| 2.     Clica no botão de “login” |
| 3.     Entra com login e senha criados no ato do cadastro. |
| 4.     Aluno loga no sistema com sucesso. |
| FLUXO SECUNDÁRIO | 4. Aluno não consegue logar no sistema com sucesso. |
| 5. Retorna aos campos de login e senha |
| 6. Revisa os dados informados. |

3.1.10 Login do pai/responsável

|  |  |
| --- | --- |
| **RF10** | |
| NOME | Login do pai ou responsável |
| DESCRIÇÃO | O sistema deve permitir que o pai ou responsável do aluno, possa fazer seu login no sistema utilizando apenas o nome de login ou e-mail e senha. |
| ATORES | Pai/Responsável |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Pai/responsável seleciona os campos e adiciona o login e a senha. |
| SAÍDAS E PÓS-CONDIÇÕES | Pai/responsável consegue logar no sistema com sucesso. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Aluno abre app/site |
| 2.     Clica no botão de “login” |
| 3.     Entra com login e senha criados no ato do cadastro. |
| 4.     Aluno loga no sistema com sucesso. |
| FLUXO SECUNDÁRIO | 4. Aluno não consegue logar no sistema com sucesso. |
| 5. Retorna aos campos de login e senha |
| 6. Revisa os dados informados. |

3.1.11 Anunciar professor (Aulas de reforço)

|  |  |
| --- | --- |
| **RF11** | |
| NOME | Anunciar professor |
| DESCRIÇÃO | O sistema deve permitir que o professor crie um tipo de “anuncio” para que a aplicação também sirva como um impulsionador ao seu negócio. |
| ATORES | Professor |
| PRIORIDADE | Alta |
| REQUISITOS NÃOFUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Professor registra seu endereço, telefone, e-mail e as aulas que ministra. |
| SAÍDAS E PÓS-CONDIÇÕES | “Anuncio” criado com sucesso. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Professor abre app/site. |
| 2.     Clica no botão de “login” |
| 3.     Entra com seu e-mail e senha |
| 4.     Clica no botão de “Anúncios” |
| 5.     Clica no botão de criar um anúncio |
| 6.     Preenche todos os campos |
| 7. Anúncio criado com sucesso |
| FLUXO SECUNDÁRIO: | 7. Anúncio não criado com sucesso. |
| 8. Professor retorna aos campos anteriores |
| 9. Revisa os dados inseridos |
| 10. Conclui a criação do anúncio. |
|  |

3.1.12 Buscar professor

|  |  |
| --- | --- |
| **RF12** | |
| NOME | Buscar professor |
| DESCRIÇÃO | O sistema deve permitir que o aluno e o responsável possam procurar professores de reforços pela redondeza. |
| ATORES | Aluno\ responsável |
| PRIORIDADE | Alta |
| REQUISITOS NÃO FUNCIONAIS ASSOCIADOS |  |
| ENTRADAS E PRÉ-CONDIÇÕES | Aluno/responsável insere o endereço e tipo de aulas que procura. |
| SAÍDAS E PÓS-CONDIÇÕES | Professor é encontrado. |
| **FLUXOS DE EVENTOS** | |
| FLUXO PRINCIPAL: | 1.     Aluno/pai abre app ou site |
| 2.     Clica no botão de “login” |
| 3.     Entra com e-mail e senha |
| 4.     Clique no botão “Encontre um professor” |
| 5.     Aluno/pai insere seu endereço |
| 6.     Professores perto da localização informada são listados. |
| FLUXO SECUNDÁRIO: | 6. Professores não são listados |
| 7. Aluno/pai revisa endereço informado |
| 8. Professores perto da localização são listados |

**3.2 Utilidade**

3.2.1 Permitir entrada via teclado e mouse

O sistema deve aceitar entradas do usuário via teclado e mouse.

3.2.2 Elementos padronizados

O sistema deve manter um padrão nos elementos da interface do sistema, como botões, cores e dimensões.

**3.3 Confiabilidade**

3.3.1 Disponibilidade

O sistema deve estar sempre disponível, caso ocorra alguma interrupção ele deve ser restaurado o mais rápido possível.

3.3.2 Política de criação de senhas

Deve haver regras para a criação de senhas do usuário, que devem ser seguidas no procedimento de criação das senhas.

3.3.3 Sigilo das informações

O sistema deve garantir o sigilo das informações cadastradas, de modo que apenas usuários com permissão possam acessar os dados do ambiente

3.3.4 Integridade dos dados

O sistema deve preservar a integridade dos dados cadastros, de maneira que apenas o professor tenha permissão para modificar as informações cadastradas.

3.3.5 Política de criação de contas de usuários

Obrigatoriamente, dois dados devem ser fornecidos: um login e uma senha.

**3.4 Desempenho**

3.4.1 Tempo de resposta

O sistema deve ter um tempo de resposta médio 2,7 segundos para concluir transações primordiais. E um índice de velocidade de 10 segundos.

3.4.2 Capacidade

O sistema deve permitir que cada professor tenha até trinta alunos por “sala de aula”.

3.4.3 Rede

O sistema precisa necessariamente da rede de internet para funcionar adequadamente.

**3.5 Suportabilidade**

3.5.1 Favorecer a orientação do usuário.

O sistema deve ter uma interface de fácil navegabilidade e que favoreça a orientação do usuário, ou seja, as opções “voltar” e “sair” devem estar presentes, de modo que o usuário não precise utilizar o navegador quando desejar retornar a uma tela ou sair do sistema. Não deve haver links quebrados e eles devem apontar exatamente para onde indicam.

3.5.2 Campos intuitivos e autoexplicativos

É importante que os campos do formulário de cadastro do Sistema sejam intuitivos e autoexplicativos o suficiente, de modo a impedir que o usuário não preencha os mesmos por não saber do que se tratam.

3.5.3 Adequação aos navegadores

Na versão web, o sistema deve ter uma interface compatível com o Internet Explore, Mozilla Firefox e Google Chrome.

3.5.4 Facilidade para recuperação de falhas

No caso de falhas, como queda de um servidor, o processo de recuperação do ambiente deve ser simples e rápido, se possível, realizado de forma automática.

3.5.5 Suporte ao usuário

O sistema deve disponibilizar um suporte técnico acessível e de fácil entendimento aos usuários.

**3.6 Restrições de Design**

Esta seção está exposta e organizada no artefato de arquitetura de software, mas segue aqui um resumo das restrições e condições primordiais do produto.

3.6.1 Restrição de modelo de arquitetura

O sistema deve ser desenvolvido de acordo com o modelo de arquitetura MVC e sempre seguindo os parâmetros das três camadas impostas por ele: Model; View; Controller.

3.6.2 Restrição de linguagens e desenvolvimento

O sistema deve ser desenvolvido em PHP e JavaScript, fazendo uso também de HTML5 e CSS para formatação.

O sistema deverá utilizar Bootstrap, visando o melhoramento da experiência do usuário e focando num design responsivo e com foco na usabilidade.

3.6.3 Restrição de banco de dados.

O sistema deverá ser modelado para banco MySQL.

**3.7 Documentação do Usuário On-line e Requisitos do Sistema de Ajuda**

O sistema de ajuda do software proposto está sendo dividido em duas partes, a primeira, em que os usuários podem visualizar as perguntas e dúvidas frequentes que podem surgir sobre o aplicativo, e a segunda, como também meios de comunicação que ligam os usuários ao suporte técnico, como o e-mail.

**3.8 Componentes Comprados**

Até o presente momento, a equipe não visou a necessidade da compra de nenhum componente, sejam eles utilizados com o sistema, ou licenças de uso e padrões associados a compatibilidade, interoperabilidade ou interface.

**3.9 Interfaces**

*3.9.1 Interfaces com o Usuário*

* Tela inicial que terá o acesso ao login do usuário, tanto professor ou responsável, também terá “Cards” que irá levar o usuário até os acessos de cadastro, anúncios e ajuda. Logo abaixo terá e-mail, números de telefone e redes sociais para que os usuários entre em contato conosco.
* Tela de cadastro a qual os usuários (Professor ou Responsável) irão colocar seus dados pessoais para ter acesso ao nosso sistema.
* Tela do anuncio de professores, onde os responsáveis irão procurar o professor para dar reforço ao seu filho.
* Tela de Ajuda a qual os usuários visualizarão as dúvidas mais frequentes do nosso sistema.
* Tela principal do professor onde terão os alunos que ele possui o vínculo, terá a tela do aluno que ele poderá anexar as atividades e colocar o desempenho dos mesmos. Existe a tela que ele cadastrará seu anuncio.
* Tela principal do responsável, onde o usuário Responsável visualiza os filhos que estão cadastro no sistema e os professores que estão vinculados a ele, na mesma tela ele pode vizualizar o desempenho do seu(s) filho(s) que o professor colocou e as tarefas anexadas. Em outra tela terá onde ele poderá cadastrar seu(s) filho(s).
* tela principal do aluno onde ele poderá visualizar as tarefas anexadas do professor.

*3.9.2 Interfaces de Hardware*

Teremos um servidor de alto desempenho que receberá os dados dos usuários e alimentará a aplicação que em tempo real processará todas as informações, os usuários poderão acessar a aplicação pelos seus dispositivos móveis ou desktops.

*3.9.3 Interfaces de Software*

Utilizaremos API de geolocalização nativa e dados de GPS para realizar as buscas de professores no momento em que o Cliente (Responsável) procurar por Professores na região mais próxima, onde exibiremos numa tela de anúncios de professores com sua posição na região de atendimento.

*3.9.4 Interfaces de Comunicações*

A aplicação utilizará a conexão com a internet para realizar comunicação entre os usuários e o servidor, também utilizaremos uma API de chat monitorada, para que os Usuários (Responsáveis e Professores) possam se comunicar e resolver qualquer dúvida rapidamente.

**3.10 Requisitos de Licença**

Nosso sistema utiliza softwares livres e Open Source, serviços de terceiros que poderão ser necessários serão devidamente licenciados.

**3.11 Observações Legais, sobre Direitos Autorais e Outras Observações**

Haverá um termo contratual na finalização do cadastro, no qual, os usuários, ao se cadastrarem tomarão conhecimento acerca dos direitos legais e licenciamento de serviços e softwares necessários para o uso do sistema e só poderá acessar a aplicação após concordar com os termos e condições.

**3.12 Padrões Aplicáveis**

O software entra em conformidade com os sistemas operacionais: Windows, Linux e Mac OS, Android e IOS.

**4. Informações de Suporte**

O produto aqui proposto pela equipe é desenvolvido com foco principal na praticidade ao usuário final, no caso, professores de reforços escolares, pais e alunos. A equipe preza pelos padrões e meios de facilitar a interação e interface do usuário, não apenas a beleza. No entanto, segue abaixo users stories, feitas com o intuito de garantir o melhor entendimento não apenas dos desenvolvedores por trás do projeto, mas também de dar suporte aos usuários.

**RF01: Cadastrar professor**

Eu, enquanto professor, quero me cadastrar no sistema para poder gerenciar o aprendizado de meus alunos de maneira mais prática e funcional. Bem como também encontrar novos clientes.

**Cenário:**

1. Professor abre o aplicativo/site
2. A tela inicial aparece
3. Clica no botão “Cadastre-se”
4. Insere os dados nos campos correspondentes a Nome, cpf, email, endereço, telefone, aulas que leciona.
5. Todos os campos são validados, cadastro concluído com sucesso.

**Cenário alternativo:**

5. Algum dos campos não é validado com sucesso e o cadastro não é concluído com sucesso.

6. Usuário retorna para os campos e revisa os dados até que todos sejam validados com sucesso.

**[RF02] cadastrar aluno**

Eu, enquanto professor, quero cadastrar meus alunos no sistema para que eles possam visualizar e também gerenciar melhor suas atividades e desempenho acadêmico.

**Cenário:**

1. Professor abre o aplicativo/site
2. Clica no botão de “login”
3. Entra com o e-mail e senha
4. Clica na aba de cadastrar aluno
5. Insere nos campos correspondentes os dados do aluno, como nome, nome dos responsáveis, e-mail dos responsáveis, endereço, telefone, telefone dos responsáveis, data de nascimento.
6. Dados são validados e o aluno é cadastrado

**Cenário alternativo:**

**6.** Dados não são validados e o aluno não é cadastrado.

7. Professor retorna aos dados informados e revisa até todos estarem corretos e sem validados, concluindo o cadastro.

**[RF005] anexar arquivos**

Eu, como professor, quero anexar materiais didáticos no sistema para que meus alunos possam, mesmo de casa, acompanhar os assuntos estudados em aula.

**Cenário:**

1. Professor abre app/site
2. Clica no botão de “login”
3. Entra com seu e-mail e senha
4. Clica na aba de alunos cadastrados
5. Escolhe o aluno
6. Clica na aba de anexos
7. Adiciona o anexo com sucesso

**Cenário alternativo:**

7.Anexo não adicionado com sucesso

1. Professor revisa formato de arquivo adicionado.

**[RF007] registrar desempenho do aluno**

Eu, como professor, quero registrar no sistema o desempenho de cada um de meus alunos. Para que assim, facilite o meu acompanhamento juntamente com os responsáveis do melhoramento desses alunos.

**Cenário:**

1. Professor abre app/site
2. Clica no botão de “login”
3. Entra com seu e-mail e senha
4. Clica na aba de alunos cadastrados
5. Escolhe o aluno
6. Clica na aba de desempenho
7. Registra o desempenho do aluno com sucesso

**Cenário alternativo:**

7. Não registra desempenho do aluno com sucesso.

8. Professor retorna aos campos de registro de desempenho

9. Revisa os dados informados.

10. Conclui o registro com sucesso.

**[RF008] Login do professor**

Eu, como professor, quero poder acessar a plataforma, tanto mobile como web, com o login que criei no fim do meu cadastro. Para que possa ter acesso as funcionalidades completas do sistema.

**Cenário:**

1. Professor abre app/site
2. Clica no botão de “login”
3. Entra com e-mail e senha criados no ato do cadastro.
4. Professor loga no sistema com sucesso.

**Cenário alternativo:**

4. Professor não consegue logar no sistema com sucesso.

5. Retorna aos campos de e-mail e senha

6. Revisa os dados informados.

**[RF009] Login do aluno**

Eu, como aluno, quero acessar a plataforma, tanto mobile como web, com o login que foi criado no fim do meu cadastro. Para que eu possa ter acesso as funcionalidades completas do sistema e acompanhar meu desempenho nas aulas de reforço.

**Cenário:**

1. Aluno abre app/site
2. Clica no botão de “login”
3. Entra com login e senha criados no ato do cadastro.
4. Aluno loga no sistema com sucesso.

**Cenário alternativo:**

4. Aluno não consegue logar no sistema com sucesso.

5. Retorna aos campos de login e senha

6. Revisa os dados informados.

**[RF010] Login do pai ou responsável**

Eu, como responsável, quero acessar a plataforma, tanto mobile como web, com o login que foi criado no fim do meu cadastro. Para que eu possa ter acesso as funcionalidades completas do sistema e acompanhar o desempenho de meu filho nas aulas de reforço.

**Cenário:**

1. Responsável abre app/site
2. Clica no botão de “login”
3. Entra com login e senha criados no ato do cadastro.
4. Responsável entra no sistema com sucesso.

**Cenário alternativo:**

4. Responsável não consegue logar no sistema com sucesso.

5. Retorna aos campos de e-mail e senha

6. Revisa os dados informados.

**[RF011] anunciar professores (aulas de reforço)**

Eu, como professor, quero que o sistema me permita criar “anúncios” e divulgar minhas aulas na plataforma. Para que assim, possa alcançar um maior número de clientes (alunos).

**Cenários:**

1. Professor abre app/site.
2. Clica no botão de “login”
3. Entra com seu e-mail e senha
4. Clica no botão de “Anúncios”
5. Clica no botão de criar um anúncio
6. Preenche todos os campos
7. Anúncio criado com sucesso

**Cenário alternativo:**

7. Anúncio não criado com sucesso.

8. Professor retorna aos campos anteriores

9. Revisa os dados inseridos

10. Conclui a criação do anúncio.

**[RF012] buscar professores (aulas de reforço)**

Eu, como aluno/pai, quero que o sistema me permita encontrar professores de reforço escolar perto de minha casa. Para que assim, facilite meu trabalho de procurar um professor de confiança.

**Cenário:**

1. Aluno/pai abre app ou site
2. Clica no botão de “login”
3. Entra com e-mail e senha
4. Clique no botão “Encontre um professor”
5. Aluno/pai insere seu endereço
6. Professores perto da localização informada são listados.

**Cenário alternativo:**

6. Professores não são listados

7. Aluno/pai revisa endereço informado

8. Professores perto da localização são listados

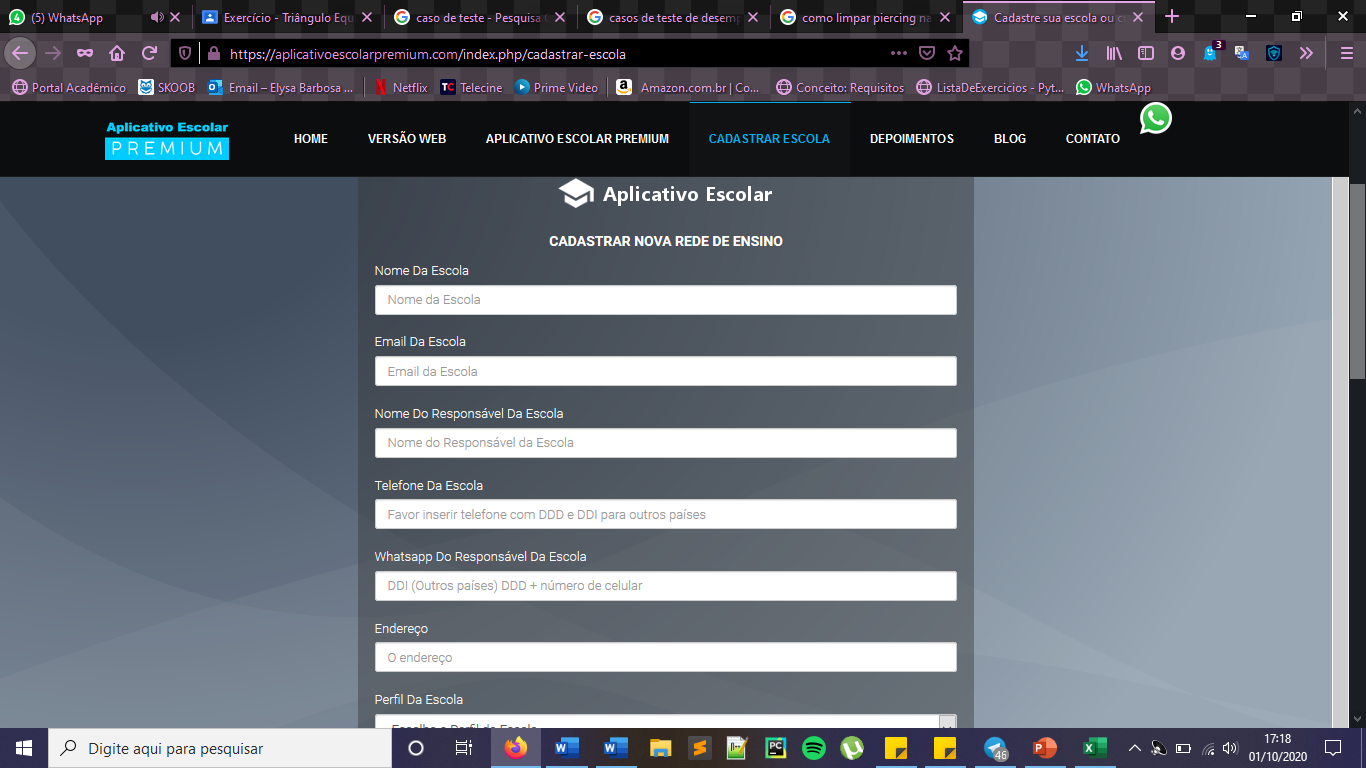
**Cenário alternativo 2:**

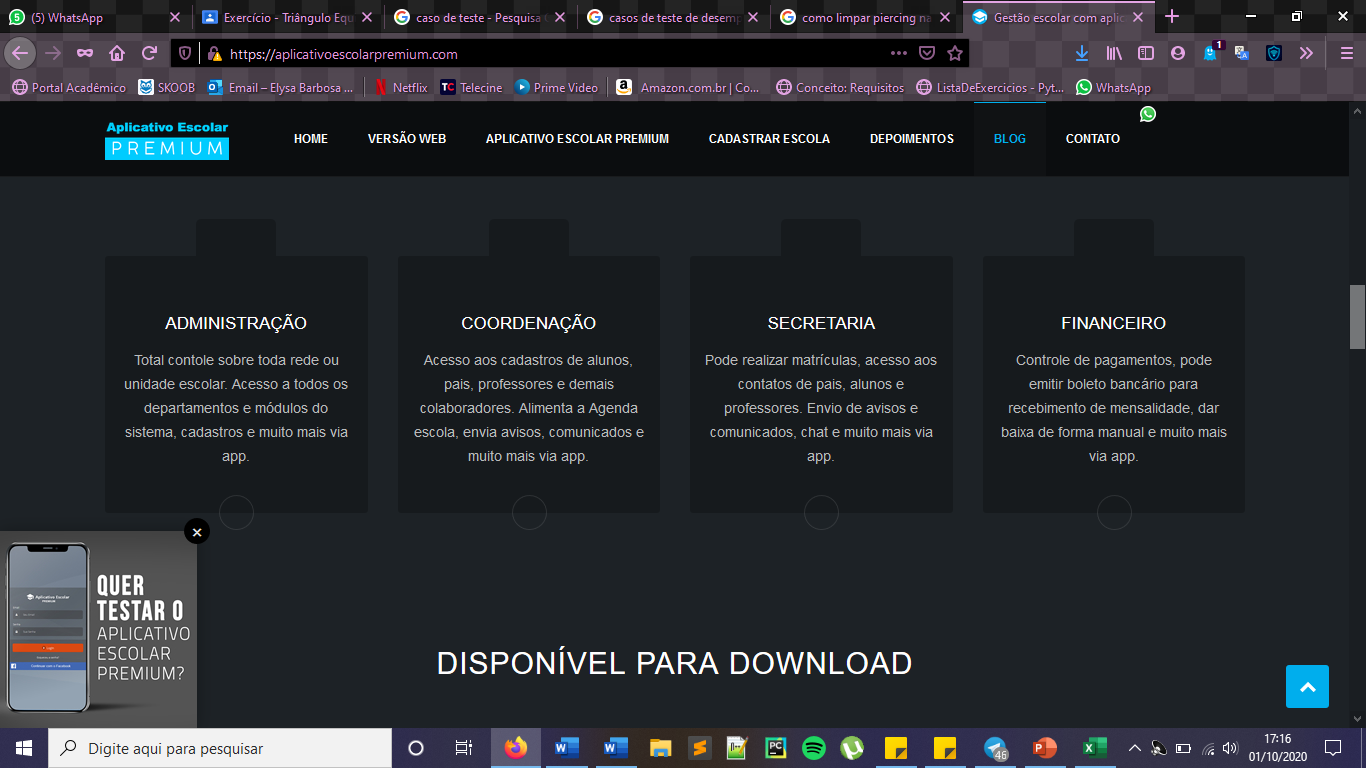
6. Não existe professores cadastrados nas redondezas do endereço inserido.

**5. Alternativas e concorrências**

* **Aplicativo Escolar**

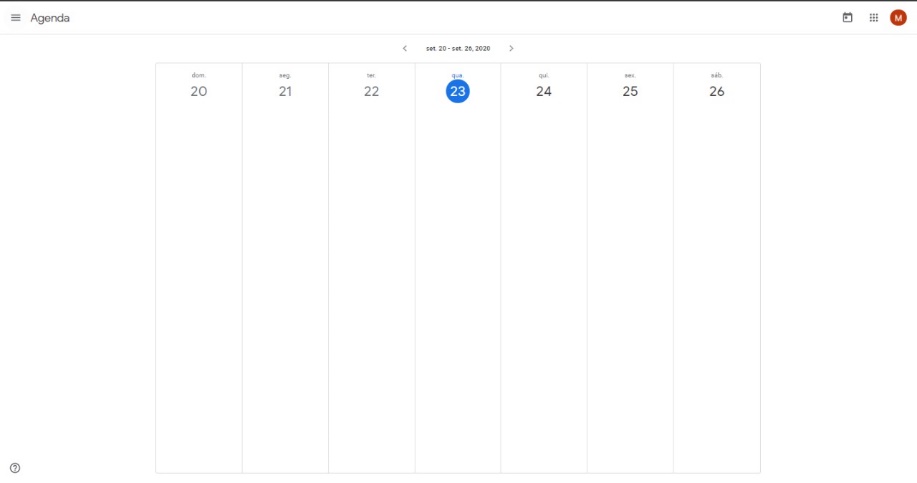
A solução funciona como um Portal do Aluno, onde a instituição cadastra-se e pode lançar notas, criar provas e testes online, consultar agenda escolar, enviar trabalhos de casa para alunos. Os pais e responsáveis/alunos também tem acesso a uma versão onde recebem comunicados e avisos da escola, realizam pagamentos de mensalidade, consultam notas, solicitam documentos, entram em contato com a coordenação. A instituição se submete ao cadastro e em seguida os gerenciadores da solução entram em contato para que possa ocorrer a contratação do serviço e a implantação da solução.

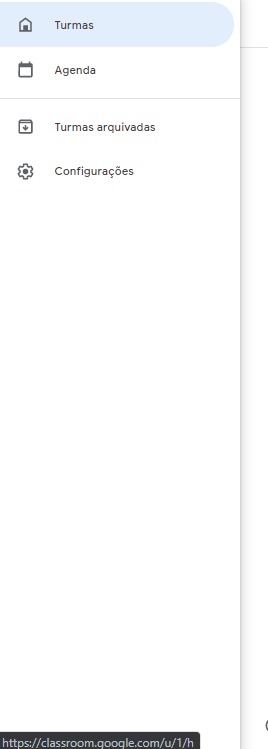




* **Google Classroom**

Com ele, o professor pode criar sua “sala de aula” virtualmente, onde também tem as funcionalidades de inserir notas, desenvolver trabalhos, dar feedback nos trabalhos enviados, criar um diário de classe, entre outras funções. Esta solução é gratuita e destinada apenas a professores e alunos cadastrados na Classroom. Pais ou responsáveis não tem um meio de visualizar estas notas, por exemplo, de uma interface separada.





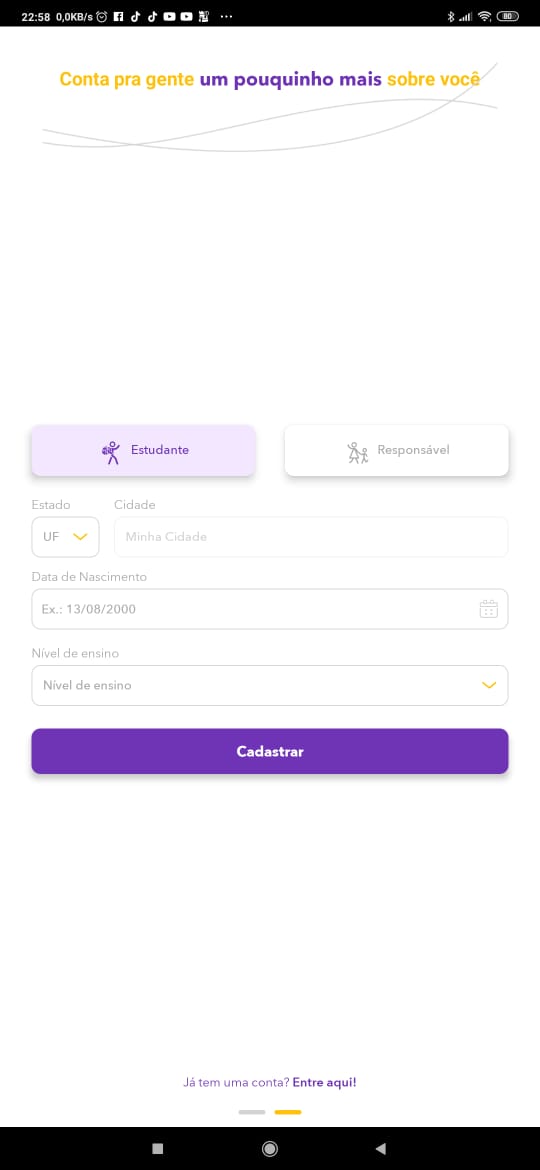
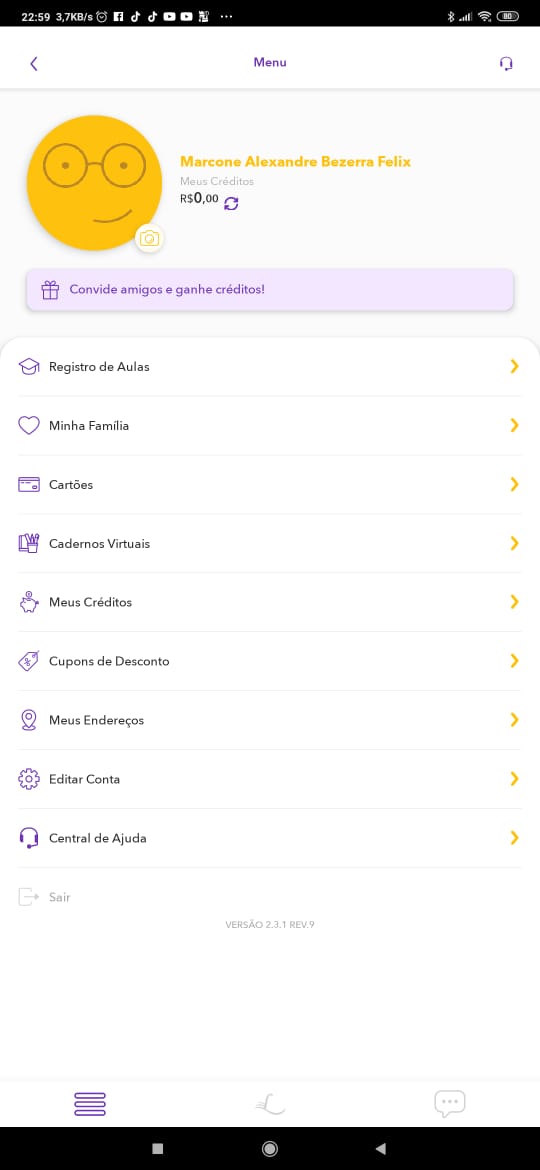
* **Eduqi**

O aplicativo é disponível para Android e IOS apenas, não consta com uma versão web utilizável. E tem como função ser um tipo de facilitador ou divulgador do trabalho do professor de reforço. Não é exclusivo para aulas de reforços escolares, nele também é possível achar aulas voltadas para universitários, idiomas e até programação, tudo com uma interface limpa e de fácil usabilidade. Com ele, o professor pode cuidar de toda a parte burocrática de seu negócio, o cliente pode até fazer o pagamento por meio do Eduqi, marcar horários para as aulas, tanto presenciais quanto online.



* **Liber**

Assim como o Eduqi, o Liber também é um aplicativo que facilita as aulas presencias e disponibiliza toda uma interface segura para garantir uma boa usabilidade nas aulas à distância. Suas funcionalidades estão disponíveis apenas para dispositivos Android e IOS.

* **EasyClasses**

Mais um da linha dos citados anteriormente, o EasyClasses também permite que professores divulguem suas aulas com uma maior praticidade, também ajuda a facilitar o pagamento dando a opção de pagamento com cartão de crédito. No entanto, a solução não tem muitas opções de professores disponíveis no Nordeste e o quesito usabilidade também chega a atrapalhar.

**6. Diferenciais**

* Sistema exclusivo para aulas de reforço escolar
* Planos mensais a preço popular
* Criação de metas para motivar os alunos
* Aulas com professores da localidade do usuário.