



**Fatec Antônio Russo**

**Projeto Integrador – AMS - ADS**

**MATEMÁTICA QUE SE VÊ**

Gustavo Almeida Carvalho  
João Gabriel Ávila Ruiz  
Nathan Moraes Adriano da Silva  
Pedro Henrique Peruzzi Vanderlei  
Yago Cauan de Oliveira da Silva

**Orientadores:**  
Carlos Henrique Veríssimo  
Jacinto Carlos Ascênsio Cansado

## Histórico da Revisão

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor(es)</b>
03/06/2022	V1	Criação do Documento	Prof. Carlos Veríssimo
10/06/2022	V2	Definição de papéis dos integrantes da equipe e modelo de negócios	Gustavo Almeida Carvalho e Pedro Henrique Peruzzi Vanderlei
24/06/2022	V3	Situação problema e tecnologias e ferramentas	Gustavo Almeida Carvalho e Pedro Henrique Peruzzi Vanderlei
31/07/2022	V4	Protótipos	Gustavo Almeida Carvalho e Pedro Henrique Peruzzi Vanderlei
16/09/2022	V5	Replanejamento dos requisitos	Gustavo Almeida Carvalho e Pedro Henrique Peruzzi Vanderlei
28/10/2022	V6	Diagramas	Gustavo Almeida Carvalho e Pedro Henrique Peruzzi Vanderlei
18/11/2022	V7	Continuação e melhorias	Gustavo Almeida Carvalho

## Identificação da Equipe

Id da Equipe: Produtos Montáveis		
Prontuário	Nome	Papel
1681432212003	Pedro Henrique Peruzzi Vanderlei	PO: Gerente de Projeto, Dev
1681432212019	Gustavo Almeida Carvalho	Scrum Master: DBA. Analista de Software
1681432212024	Nathan Moraes Adriano da Silva	Dev
1681432212026	João Gabriel Ávila Ruiz	Dev: Web
1681432212010	Yago Cauan de Oliveira Silva	Dev

## Índice

<b>1</b>	<b>DESCRIÇÃO GERAL DO CLIENTE .....</b>	<b>5</b>
1.1	DESCRIÇÃO DA NECESSIDADE .....	5
1.2	OBJETIVO.....	5
1.3	ESCOPO.....	5
<b>2</b>	<b>DESCRIÇÃO DOS GESTORES E DOS USUÁRIOS.....</b>	<b>6</b>
<b>3</b>	<b>REQUISITOS DO CLIENTE .....</b>	<b>6</b>
3.1	REQUISITOS FUNCIONAIS.....	6
3.2	REQUISITOS NÃO FUNCIONAIS .....	7
<b>4</b>	<b>SITUAÇÃO PROBLEMA.....</b>	<b>8</b>
4.1	IDENTIFICAÇÃO DO PROBLEMA .....	8
4.2	PÚBLICO-ALVO.....	8
4.3	MACRO NECESSIDADES.....	8
<b>5</b>	<b>ARQUITETURA DO SISTEMA .....</b>	<b>9</b>
5.1	MODELO DE NEGÓCIO.....	9
5.2	MAPA DE SITE .....	10
5.2.1	Web.....	10
5.2.2	Mobile .....	10
5.3	DIAGRAMAS ENTIDADE RELACIONAMENTO.....	11
5.4	DIAGRAMA DE CLASSES .....	12
5.5	DIAGRAMAS DE ATIVIDADE.....	13
5.6	CASOS DE USO.....	14
<b>6</b>	<b>TECNOLOGIAS E FERRAMENTAS.....</b>	<b>15</b>
6.1	GERENCIAMENTO DO PROJETO .....	15
6.2	PLATAFORMA.....	15
6.2.1	Mobile .....	15
6.2.2	Web.....	15
6.3	VERSIONAMENTO E IMPLANTAÇÃO.....	15
<b>7</b>	<b>PROTÓTIPOS .....</b>	<b>16</b>
7.1	WEB.....	16
7.2	MOBILE .....	16
<b>8</b>	<b>CONTINUAÇÃO E MELHORIAS.....</b>	<b>18</b>

## 1 Descrição Geral do Cliente

### 1.1 Descrição da Necessidade

A necessidade	<i>Apoio em tarefas acadêmicas e administrativas relativas à disciplina</i>
Afeta	<i>Docentes e discentes da Fatec</i>
O seu impacto é	<i>Condução da disciplina, do ponto de vista acadêmico e administrativo</i>
Benefícios com a solução	<i>Proporcionar ao docente a utilização mais otimizada do tempo; maior transparência na relação docente/discente; melhora na comunicação entre docente/discente</i>

### 1.2 Objetivo

*Proporcionar a demonstração prática de conceitos matemáticos*

### 1.3 Escopo

*Gestão de fórmulas algébricas*

*Mapear os passos de cada demonstração*

*Proporcionar fases no processo de aprendizado (performance do aluno)*

*Análise de resultado*

## 2 Descrição dos Gestores e dos Usuários

Representante: *Profa. Adriane*  
Função/Unidade: *Diretora da Fatec São Caetano do Sul*  
Papel: *Stakeholder Principal*

Representante: *Prof. Jacinto*  
Função/Unidade: *Cooordenador do curso ADS – Fatec São Caetano do Sul*  
Papel: *Stakeholder*

Representante: *Prof. Ayrton Barboni*  
Função/Unidade: *Professor de matemática da Fatec São Paulo*  
Papel: *Cliente*

Representante: *Prof. Carlos Veríssimo*  
Função/Unidade: *Professor da disciplina Técnicas Avançadas de Programação Web e Mobile*  
Papel: *Cliente*

## 3 Requisitos do Cliente

*Deverão ser desenvolvidas funcionalidades para atender às seguintes necessidades de negócio:*

### 3.1 Requisitos Funcionais

RF1: Manutenção das fórmulas algébricas

RF2: Para cada demonstração gráfica, mostrar os conceitos matemáticos envolvidos

RF3: Prever níveis de maturidade do aluno

RF4: Proporcionar análise de desempenho dos alunos

RF5: Sistema deve mostrar o desempenho do usuário a ele

### 3.2 Requisitos Não Funcionais

- RNF1:** Deverá atender às plataformas Web e Mobile
- RNF2:** Deverá ser projeto utilizando a unidade Fatec São Caetano do Sul, poré com vista à atender à todas as unidades da Fatec (Nível Estadual)
- RNF3:** Deverá ser desenvolvida um back-end robusto para dar sustentação (negócio) ao front-end (Mobile)
- RNF4:** Todas as funcionalidades de CRUD deverão ser desenvolvidas na plataforma Web
- RNF5:** Design de ícones familiares com seu uso, para reconhecimento imediato.
- RNF6:** Terá uma interface de fácil uso tanto para usuários experientes como para leigos.
- RNF7:** O desenvolvimento do software deverá prever entregas parciais para o cliente do produto de software

## **4 Situação Problema**

Ponto de partida para realização do projeto, através de uma análise detalhada do problema, público-alvo e macro necessidades

### **4.1 Identificação do problema**

O cliente deseja implementar um novo paradigma para se ensinar produtos notáveis. Ele possui um modelo físico e pretende virtualizá-lo de forma que mais alunos possam aprender o objeto de estudo.

### **4.2 Público-alvo**

Estudantes do ensino fundamental e médio, tendo em vista o contato dos alunos na grade curricular de matemática no sistema de educação do Brasil.

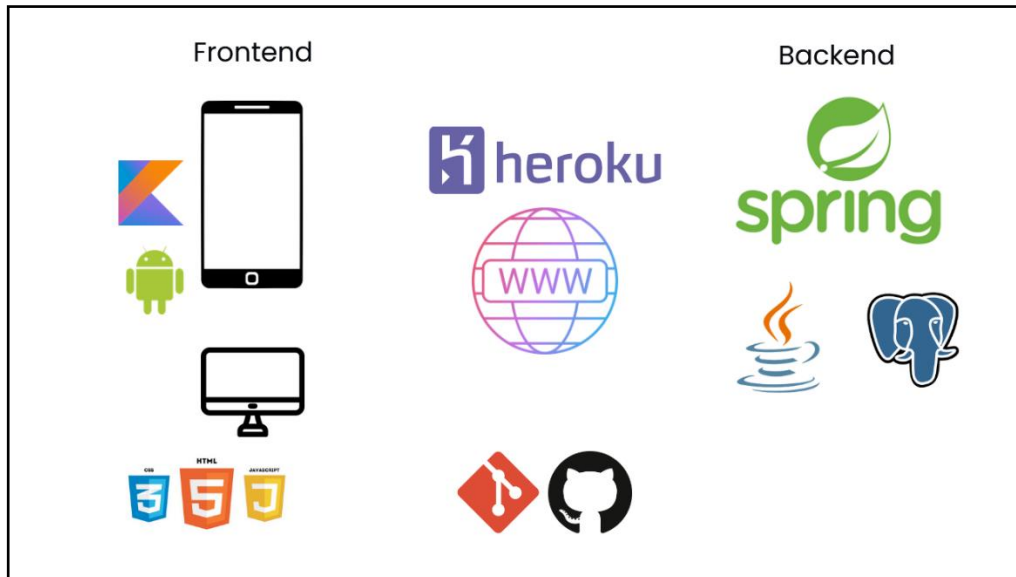
### **4.3 Macro necessidades**

O sistema deve ser capaz de fornecer ao usuário uma experiência visual e interativa, incentivar e auxiliar o aluno a resolver os desafios na plataforma, de modo que ele seja capaz de abstrair o conteúdo de produtos notáveis à resolução de problemas matemáticos.

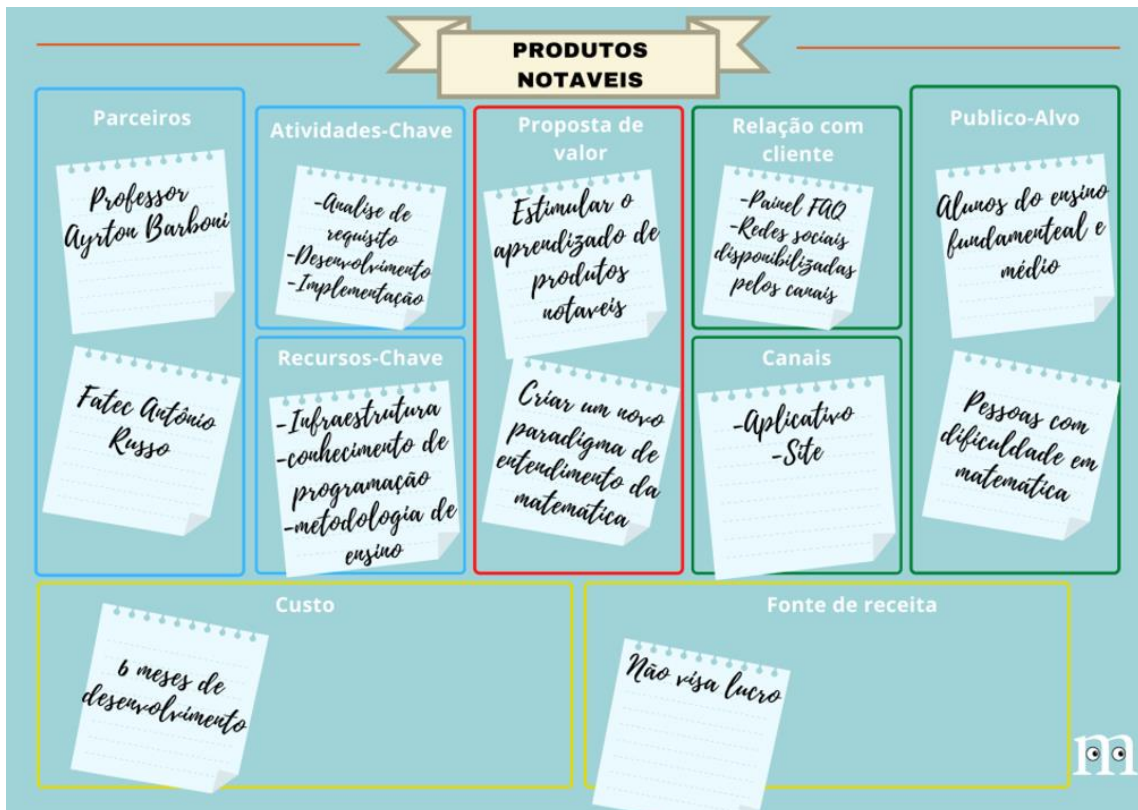


## 5 Arquitetura do Sistema

A arquitetura do sistema compreende a descrição de alto nível da solução implementada ao sistema, onde estão dispostos os diagramas elaborados para a implementação dos requisitos do cliente.

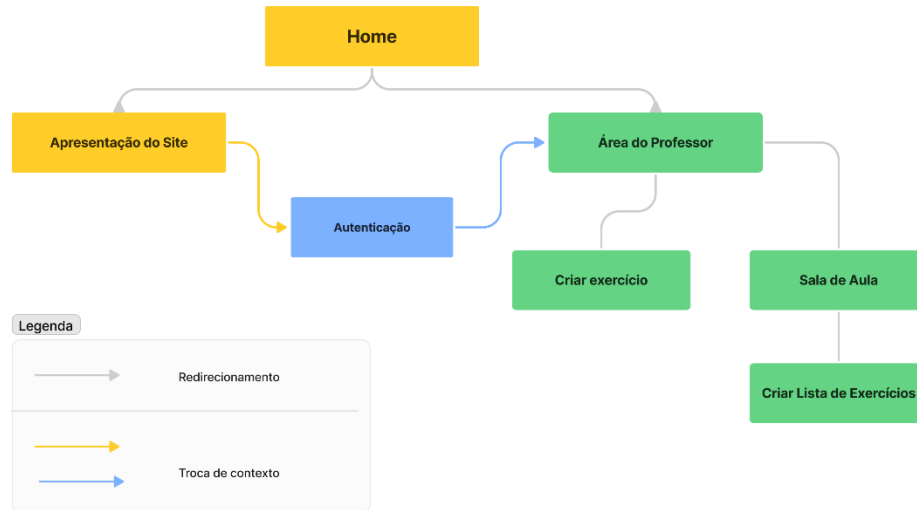


### 5.1 Modelo de Negócio

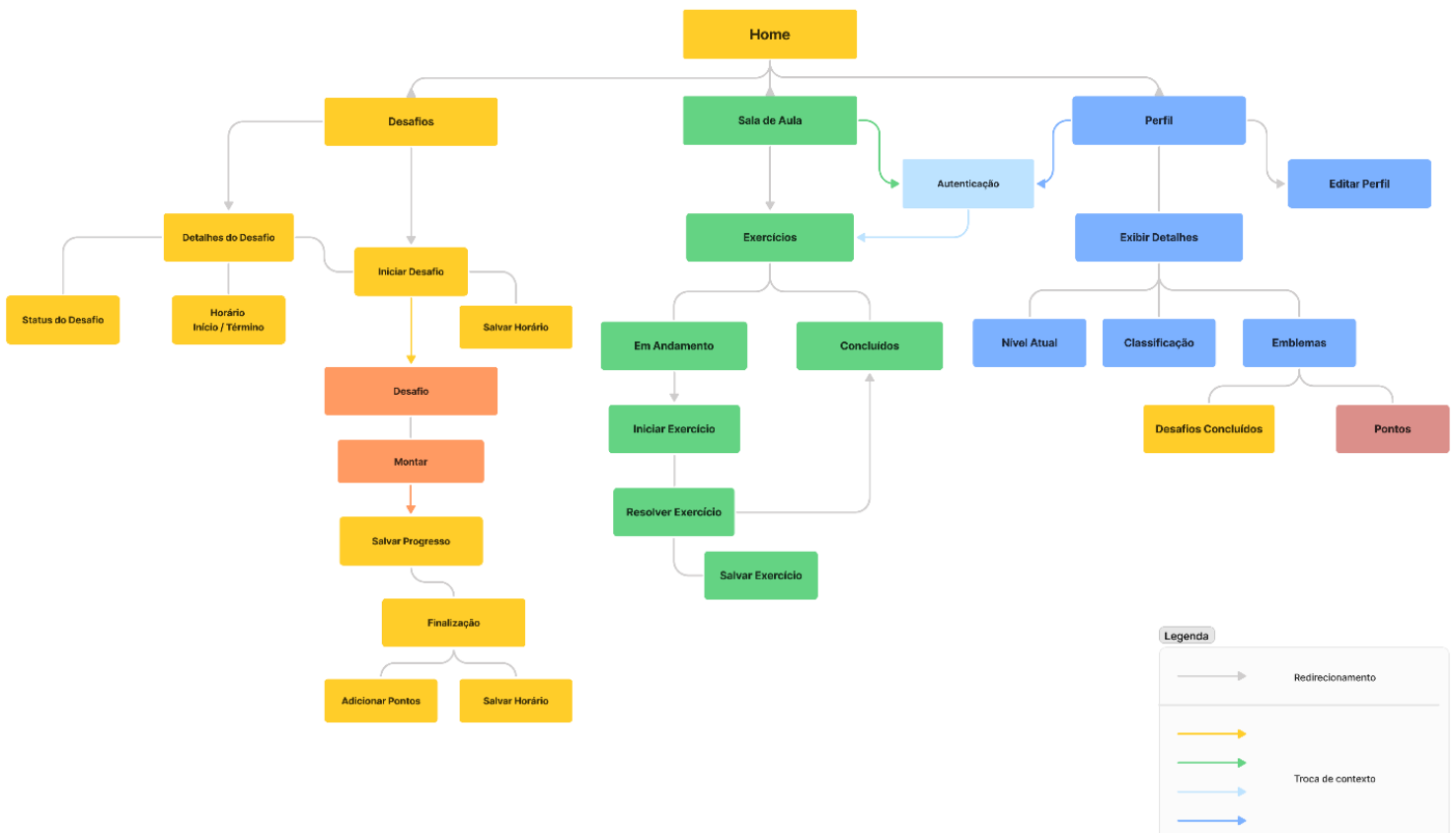


## 5.2 Mapa de Site

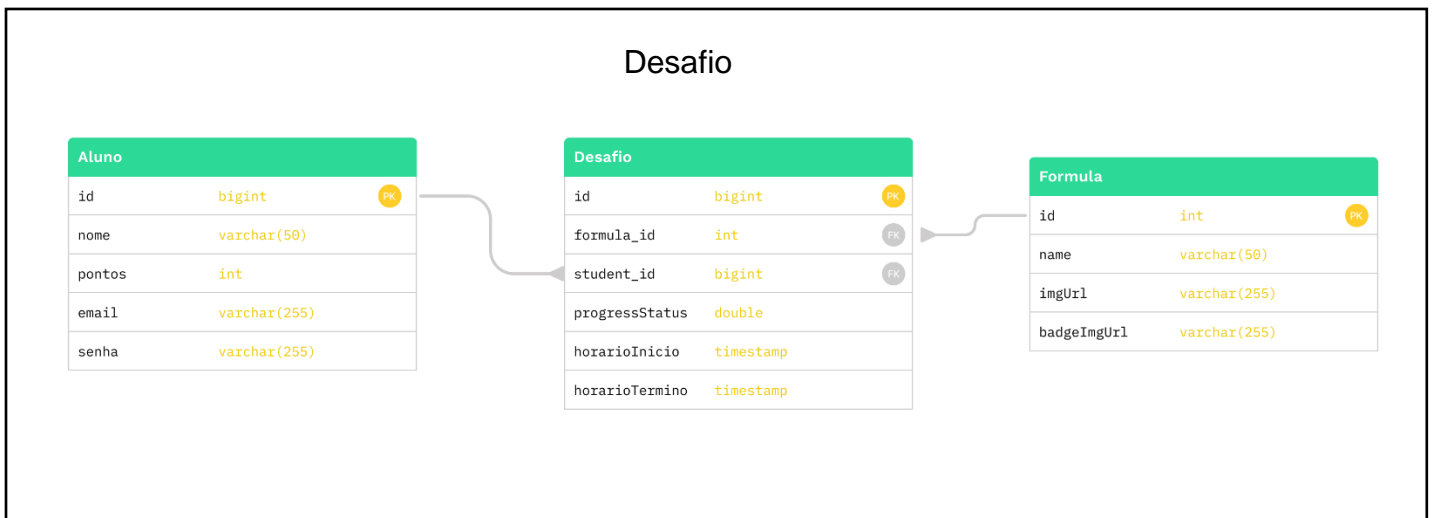
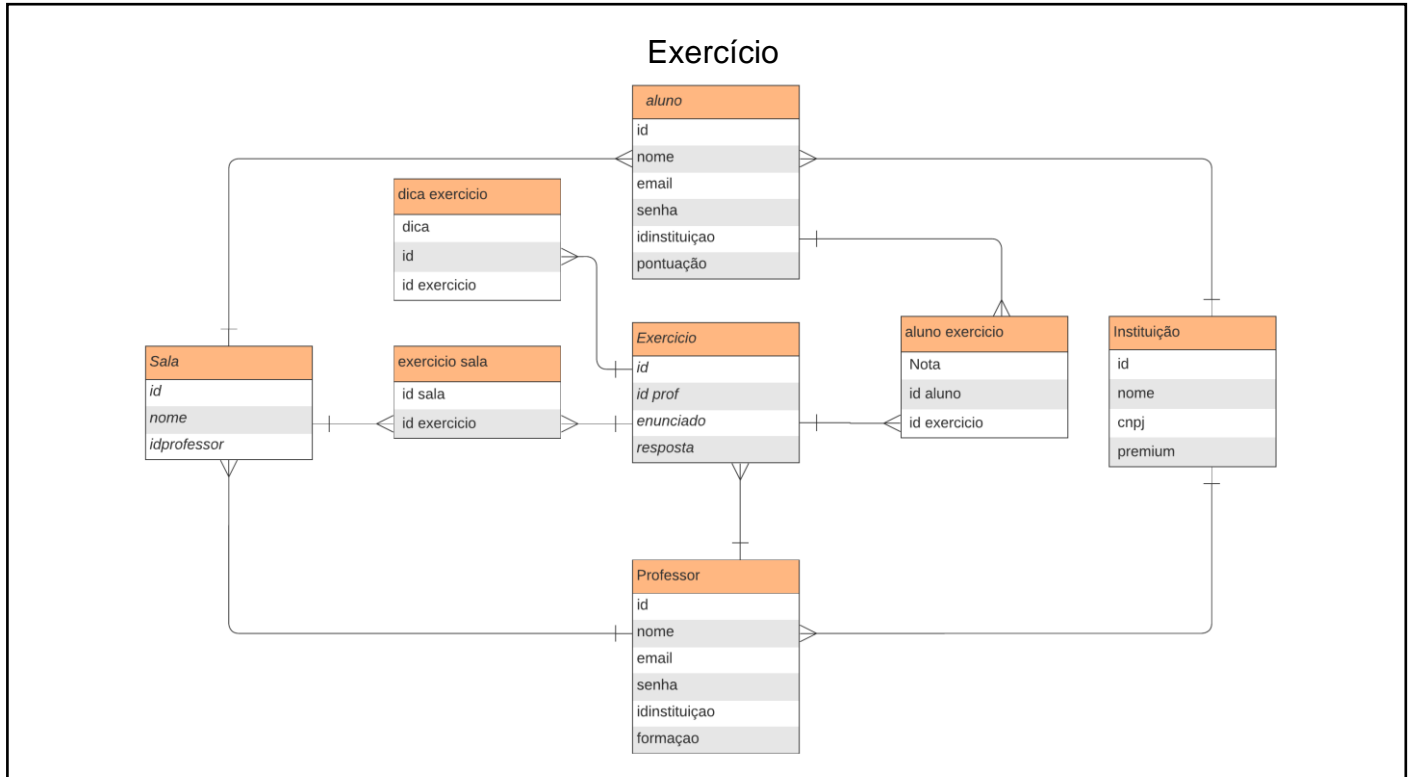
### 5.2.1 Web



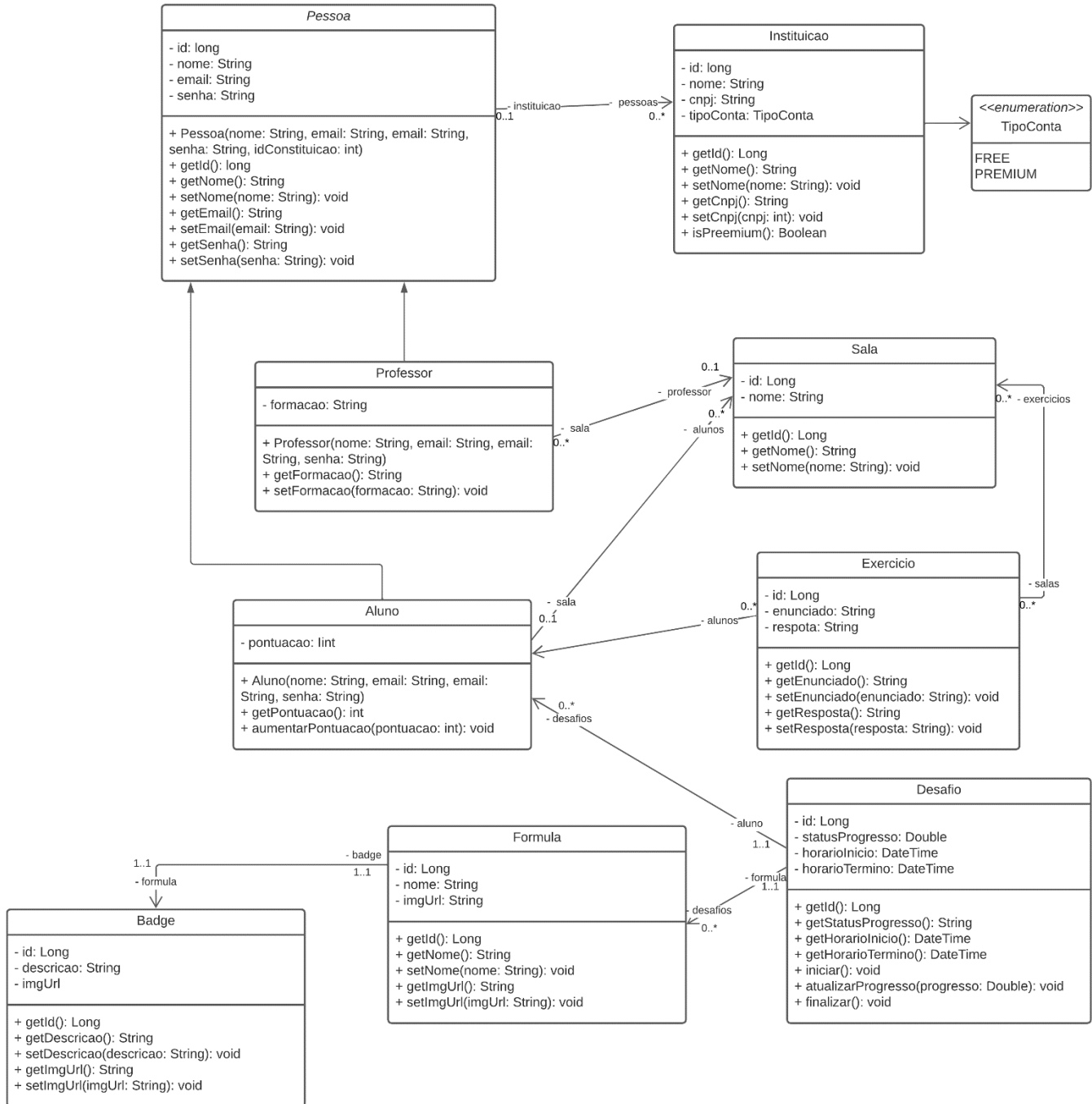
### 5.2.2 Mobile



### 5.3 Diagramas Entidade Relacionamento

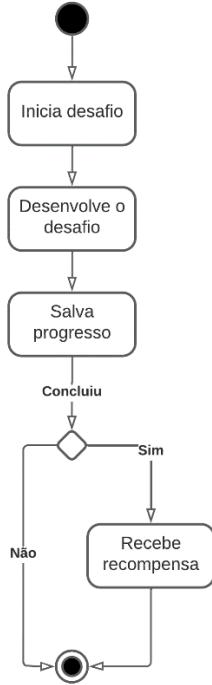


## 5.4 Diagrama de Classes

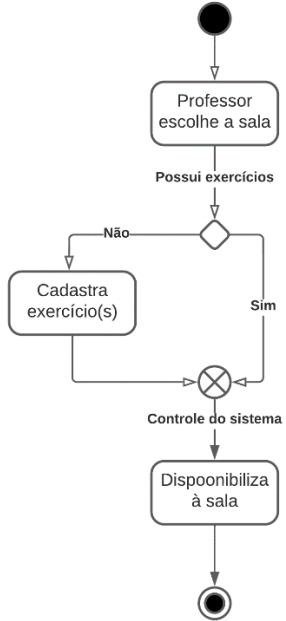


## 5.5 Diagramas de Atividade

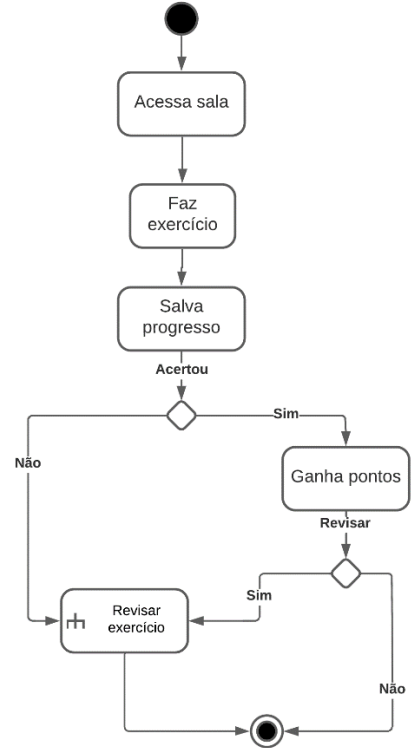
Realização dos desafios



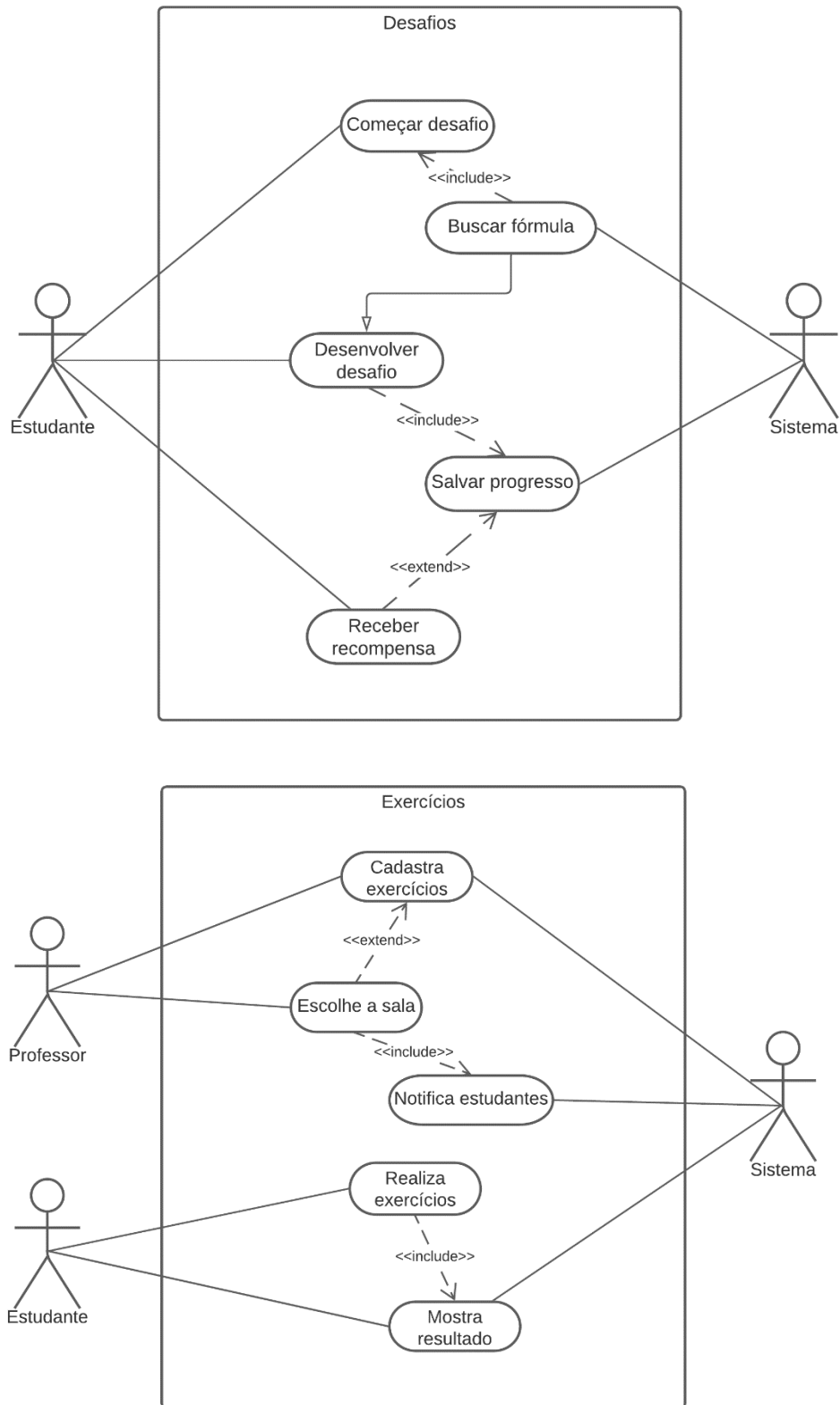
Cadastro dos exercícios



Realização dos exercícios



## 5.6 Casos de Uso



## 6 Tecnologias e Ferramentas

O conjunto de tecnologias e ferramentas utilizadas (*Tech Stach*) no projeto foi escolhido cuidadosamente de acordo com a necessidade do projeto, assim como leva em consideração a familiaridade e experiência dos integrantes da equipe.

### 6.1 Gerenciamento do projeto

O Trello foi usado para gerenciamento do projeto através, onde as atribuições são definidas em um quadro seguindo a metodologia do Scrum. Para a comunicação entre os membros da equipe, assim como realização das Sprints, foi utilizada a plataforma Discord.

### 6.2 Plataforma

Refere-se ao ambiente onde o software é executado.

#### 6.2.1 Mobile

O software terá como foco o desenvolvimento em Android, utilizando a ferramenta Android Studio, visto que grande parte da população brasileira possui um aparelho celular, sendo o Android o sistema operacional de aparelhos celulares no Brasil.

#### 6.2.2 Web

A plataforma web será utilizada como serviço para o aplicativo. O serviço consiste em uma *API* no padrão *REST*. Para isso, o software utilizado é desenvolvido em Spring Boot através da plataforma IntelliJ IDEA Community.

### 6.3 Versionamento e Implantação

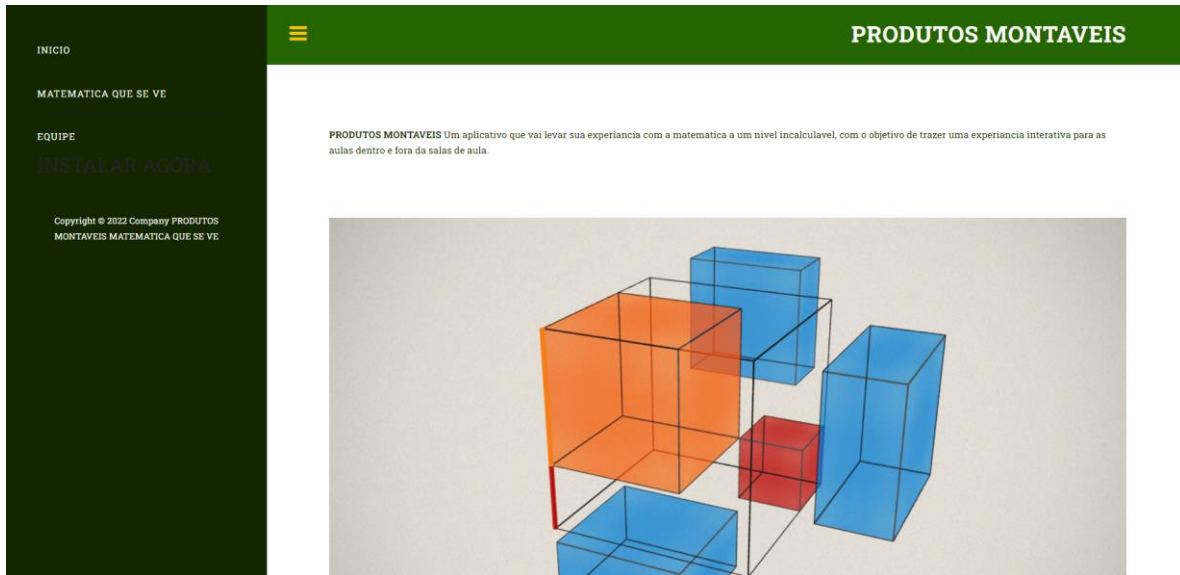
GitHub foi utilizado para o versionamento dos códigos-fonte do aplicativo e Web APIs. O website foi hospedado no serviço do GitHub Pages.

A plataforma utilizada para a implantação das APIs é o Heroku, uma vez que oferece serviço de implantação gratuito para pequenos projetos, além de fácil integração com o GitHub.

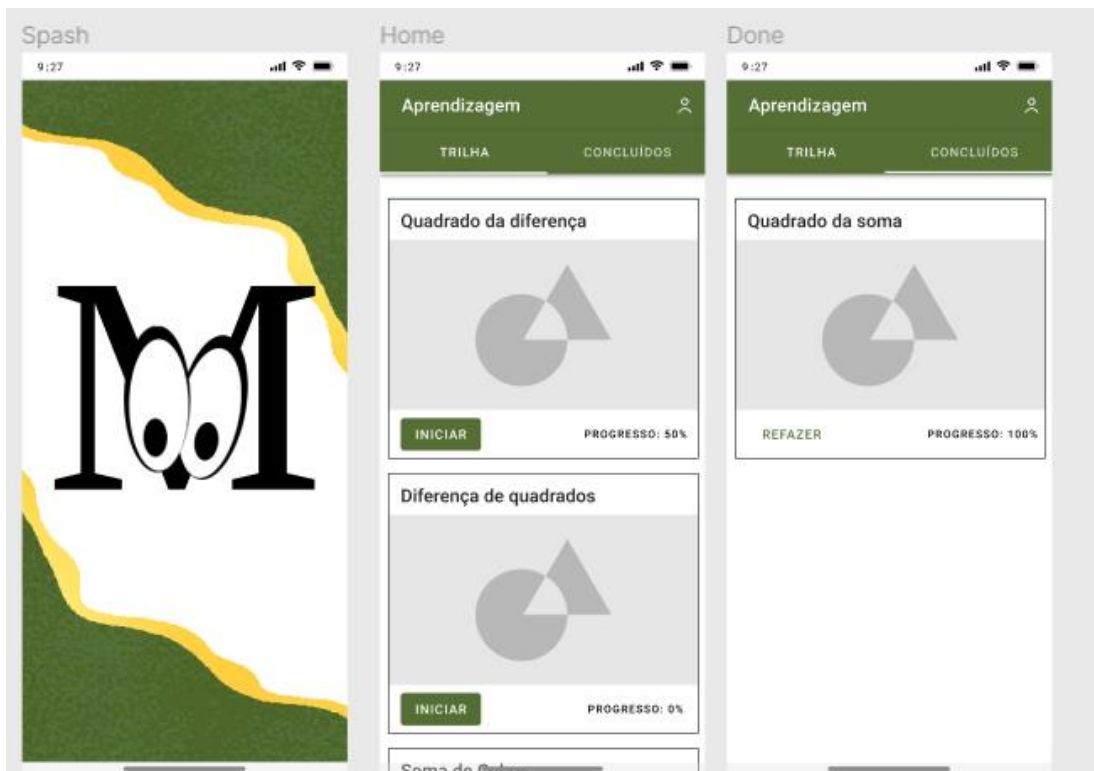
## 7 Protótipos

Seção designada para a demonstração visual entre layouts e relacionamento das interfaces, desenvolvida através da plataforma Figma.

### 7.1 Web



### 7.2 Mobile





## A matemática que se vê



## 8 Continuação e Melhorias

O projeto tem como objetivo a apresentação de um MVP previsto para o final do ano letivo de 2022 para composição curricular da disciplina. Dessa forma, a equipe se compromete a realizar a entrega dos requisitos especificados nesse documento. No entanto, o produto está passível a mudanças para manutenções evolutivas, a lista abaixo consta alguns pontos de melhoria para a pós entrega do MVP.

### 8.1 Funcionalidades

1. Implementação de mais desafios interativos em 3D (fora do alcance da equipe);
2. Permitir que o aluno realize o mesmo desafio várias vezes;
3. Salvar todas as tentativas de cada desafio;
4. Exibir na aba principal, a melhor pontuação;
5. Cronometragem de conclusão do desafio;
6. Associação do professor a instituições;
7. Implementar sistema de anúncios;
8. Conexão com conta Google.

### 8.2 Arquitetura

Com o cancelamento do plano gratuito do Heroku, percebe-se uma necessidade de uma plataforma mais adequada para hospedagem da aplicação, de modo que não traga custos à equipe, ao mesmo tempo que seja de fácil escalonamento caso o projeto venha a ser continuado como uma opção mais acessível que o Heroku.

- **Website:** O site do projeto está hospedado no GitHub Pages atualmente. Enquanto essa solução é plausível para pequenos projetos, existem outras plataformas mais adequadas para aplicações front-end. Dentre elas, algumas plataformas que oferecem os serviços completamente gratuitos são:

**Netlify:** <https://www.netlify.com/>

**Vercel:** <https://vercel.com/>

Todas as opções acima fornecem um serviço completamente gratuito, com possibilidade de evolução para um plano pago, como vantagens como DNS personalizado e menor limitação dos recursos oferecidos.

- **API:** As APIs do projeto estão hospedadas na plataforma Heroku. Algumas soluções para hospedagem das APIs que podem substituir a atual plataforma de forma completamente gratuita são:

**Render:** <https://render.com/>

**Cyclic:** <https://www.cyclic.sh/>

Ambas as plataformas permitem hospedagem completamente gratuita de uma API, no entanto, a PaaS Cyclic, apesar de fornecer solução fullstack, apenas permite hospedagem de aplicações em JavaScript.

- **Banco de Dados:** O SGDB utilizado no projeto é o Postgres. O Heroku costumava prover um banco de dados gratuito para pequenas aplicações. Alguns provedores de bancos de dados possíveis que não trariam custos durante a fase inicial após o MVP:

**FreeSql:** <https://www.freesqldatabase.com/>

**WebHost:** <https://www.000webhost.com/>

**HelioHost:** <https://heliohost.org/>

**Cockroach:** <https://www.cockroachlabs.com/>

Todas as opções citadas oferecem um plano gratuito, portanto candidatos possíveis para o projeto. A plataforma FreeSql e HebHost oferecem o serviço para o SGDB MySql e seus principais usos são para pequenas aplicações, a plataforma HelioHost oferece o serviço para Postgres e o sistema é gratuito sustentado por doações, enquanto a CockroachLabs oferece seu próprio SGDB, Cockroach, que oferece um serviço gratuito limitado que pode ser evoluído para um plano com mais recursos.

- **API + Banco de dados:** Existem soluções para hospedagem tanto das APIs, quanto do banco de dados. Dentre elas.

**Railway:** <https://railway.app/>

**Caprover:** <https://caprover.com/>

Ambas as opções fornecem o serviço de forma gratuita. A plataforma Railway oferece serviço freemium, onde os serviços possuem limitações, mas não apenas o teste é gratuito, sendo ideal para pequenos projetos fullstack. A Caprover é uma plataforma open source e os serviços de hospedagem são gratuitos. No entanto, sua implementação é feita através de contêineres, tecnologia que a equipe não possui conhecimento para implementação, mas se dispõe como futuro candidato.

### Observação:

Provedores de cloud como AWS, Azure, GCP e Digital Ocean oferecem grande parte dos serviços gratuitos para teste, mas são ótimas opções para aplicações que exijam alta disponibilidade e uma equipe capaz de gerenciá-las.

## A matemática que se vê