CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE CAMPINAS

CURSO SUPERIOR DE TECNOLOGIA EM GESTÃO DA TECNOLOGIA DA INFORMAÇÃO

ANDRÉ CARVALHO ZANCHETTA

DAVID AQUINO QUEIROZ

**APLICATIVO MOBILE COM FUNCIONALIDADES DO SIGA**

 CAMPINAS/SP

2024-1ºsem

**1. REQUISITOS E DIAGRAMA DE CASO DE USO**

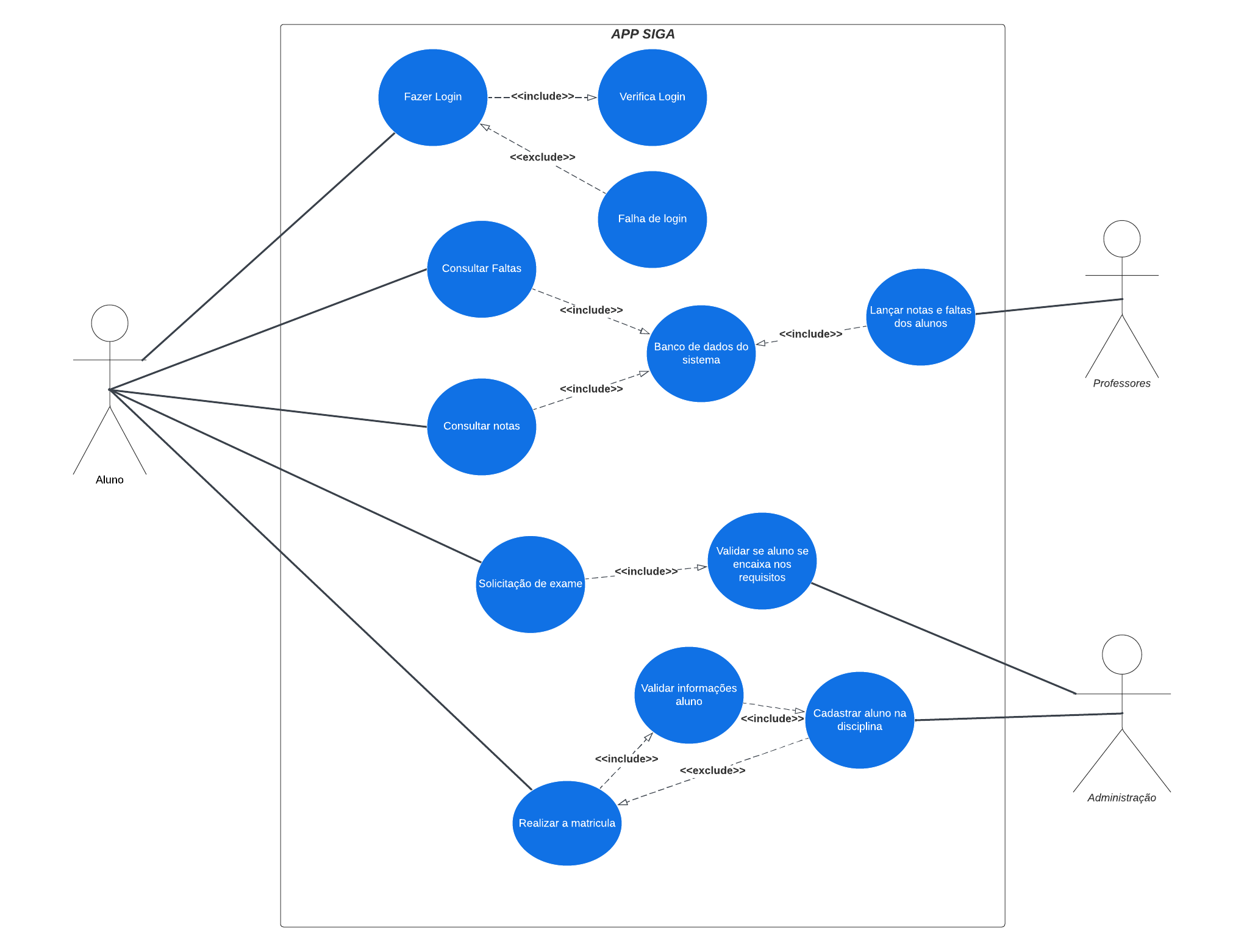
* 1. **Requisitos funcionais**

|  |  |
| --- | --- |
| RF - 01 | O aplicativo deve realizar a matrícula dos alunos nas disciplinas. |
| RF - 02 | O aplicativo deve mostrar e armazenar as notas dos alunos lançadas pelo professor e realizar o cálculo automaticamente da fórmula estipulada. |
| RF - 03 | O aplicativo deve disponibilizar para os alunos consultar as faltas nas devidas disciplinas de forma transparente e clara. |
| RF - 04 | O aplicativo deve permitir que os alunos e professores pesquisem, reservem e renovem livros, além de gerenciar empréstimos e devoluções |
| RF - 05 | O aplicativo deve ter uma aba para solicitações de exame para o Aluno. |
| RF - 06 | O aplicativo deve exibir as disciplinas matriculadas de cada aluno e suas respectivas ementas. |
| RF - 07 | No aplicativo será possível reservar laboratórios, equipamentos e outros recursos necessários para aulas práticas e atividades de pesquisa. |
| RF - 08 | O aplicativo deverá emitir e gerenciar certificados, diplomas e outros documentos oficiais para os alunos que completarem com sucesso seus cursos e programas. |
| RF -09 | O aplicativo deve ser capaz de cadastrar, atualizar e remover informações dos alunos, como matrícula, notas, histórico acadêmico etc. |

* 1. **Requisitos não funcionais**

|  |  |
| --- | --- |
| RNF - 01 | O aplicativo deve rodar em sistemas Android e IOS. |
| RNF - 02 | O aplicativo deve ter acesso de login com CPF e senha do usuário. |
| RNF - 03 | O aplicativo deve ser intuitivo e rodar de forma clara. |
| RNF - 04 | O aplicativo deve ser capaz de lidar com um aumento no número de usuários e no volume de dados ao longo do tempo, sem que isso afete negativamente o desempenho. |
| RNF - 05 | O aplicativo deve garantir que dados dos alunos, como informações pessoais, notas e registros acadêmicos, estejam protegidos contra acesso não autorizado. |
| RNF - 06 | O aplicativo deve ser confiável, minimizando falhas e erros inesperados que podem atrapalhar as atividades dos alunos. |
| RNF - 07 | No aplicativo será possível reservar laboratórios, equipamentos e outros recursos necessários para aulas práticas e atividades de pesquisa. |
| RNF - 08 | O aplicativo deverá fazer backups de rotina. |
| RNF - 09 | O aplicativo deve ser responsivo para rodar bem em várias resoluções como celular ou tablet. |
| RNF - 10 | O aplicativo deve conter todas as atividades do usuário no sistema são registradas para auditoria e rastreamento, garantindo transparência e responsabilidade. |

* 1. **Diagrama de caso de uso**



* 1. **Atores**
* **Alunos:** Os alunos são os usuários finais do aplicativo, sendo o maior afetado no uso do aplicativo, acessando informações como por exemplo notas, horários e as disciplinas matriculadas.
* **Professores:** Irão utilizar os sistemas para lançar presenças e notas para os alunos, e verificar quais são as turmas que ele ministra.
* **Administração Acadêmica:** Irá utilizar para verificar as questões das disciplinas cadastradas de cada aluno, a administração acadêmica terá um treinamento para prestar um suporte para os usuários das devidas funções do aplicativo.

**1.5 Narrativas de caso de uso**

* **Matrícula do Aluno**

**Ator Principal:** Aluno

**Pré-condições:** O aluno precisa estar logado no aplicativo

**Fluxo Principal:**

1. O aluno acessa a aba de matrícula no sistema;
2. Sistema exibe as disciplinas disponíveis para matrícula;
3. Aluno escolhe as disciplinas e faz a matrícula;

**Fluxo Alternativo:**

* Se o aluno já estiver matriculado nas disciplinas ele pode consultá-las.
* **Lançamento e Consulta de Notas**

**Ator Principal:** Professor

**Pré-condições:** O professor precisa ter acesso ao aplicativo e permissão

**Fluxo Principal:**

1. O professor acessa o sistema e seleciona a opção de lançamento de notas.
2. Sistema lista as disciplinas ministradas pelo professor e permite que ele escolha uma delas;
3. Professor insere notas;
4. Ficará disponível para os alunos visualizarem as notas;

**Fluxo Alternativo:**

* O professor pode editar as faltas caso já tenha realizado a marcação das faltas.
* **Consulta de Faltas**

**Ator Principal:** Aluno

**Pré-condições:** Os Alunos precisam ter acesso ao aplicativo

**Fluxo Principal:**

1. O Aluno entra no aplicativo e acessa a aba de consultar notas;
2. O aplicativo exibe a lista de disciplinas do aluno e suas respectivas faltas;

* **Solicitação de exame**

**Ator Principal:** Aluno

**Pré-condições:** Os Alunos precisam ter acesso ao sistema.

**Fluxo Principal:**

1. O Aluno entra no aplicativo e acessa a aba de solicitações de exame;
2. O Aluno seleciona a matrícula que gostaria de fazer o exame (semestre passado);
3. Aguarda confirmação da secretária acadêmica;
   1. **Ferramentas utilizadas**

* **Lucidchart:** O Lucidchart é uma plataforma online que permite aos usuários criarem, colaborar e compartilhar diagramas e fluxogramas de forma fácil e intuitiva, com isso utilizamos para desenhar os diagramas de caso de uso do SOPA.

1. **INTERFACE**

**2.1 Proposta de Interface**

Para criar uma proposta de interface que seja realmente funcional e atraente, focamos em adotar um design moderno, seguindo as tendências atuais. Nosso objetivo é desenvolver uma narrativa visual que seja clara, intuitiva e agradável para o usuário final. Pensamos em todos os detalhes, desde a escolha das cores e fontes até a disposição dos elementos na tela. Queremos que o usuário se sinta confortável e encontre o que precisa de maneira rápida e fácil. Para isso, aplicamos as melhores práticas de *UX* e *UI Design*, garantindo um espaçamento adequado, boa legibilidade e uma navegação fluida.

**2.2 Ferramentas**

Para desenvolver o design da interface, utilizamos o Figma, uma ferramenta poderosa que nos permitiu criar protótipos interativos e colaborar em tempo real. Isso facilitou muito nosso trabalho, já que podíamos compartilhar ideias e ajustar detalhes rapidamente.

Link do projeto:<https://www.figma.com/design/WqEcByXtIUCaCgByHAaUc4/SIGA-Mobile?node-id=0-1&t=HBYbJwRsMb4tU26y-1>

1. **GESTÃO DE CONFIGURAÇÃO DE SOFTWARE**

Para gerenciar o código do nosso projeto, escolhemos utilizar o Github. Essa plataforma é ótima porque nos permite manter o controle de todas as alterações e colaborar de maneira eficiente. Além disso, usamos o GIT como tecnologia de versionamento de código. Com isso, conseguimos rastrear todas as modificações, reverter mudanças se necessário e trabalhar em equipe sem problemas.

Link do projeto: <https://github.com/Dnowdd/siga-mobile>

1. **METRICAS**

**4.1 Pontos de função**

Baseado nos requisitos funcionais fizemos um breve cálculo de pontos de funções, conforme abaixo (consideramos todos como prioridade baixa).

EE CADASTRAR USUARIO – 6 TEDS – 1 TAR

EE ALTERAR DADOS USUARIO – 4 TEDS – 1 TAR

EE BLOQUEAR CADASTRO – 1 TEDS – 1 TAR

EE MATRÍCULA DO ALUNO – 5 TEDS – 2 TAR

EE LANÇAR FALTA – 3 TED – 2 TARS

EE LANÇAR FALTA – 3 TED – 2 TARS

CE CONSULTAR NOTA PELO ALUNO – 2 TED – 1 TAR

CE CONSULTAR FALTA – 2 TED – 1 TAR

SE EMISSÃO DE CERTIFICADOS – 4 TEDS – 3 TARS

Segue abaixo tabela utilizado como exemplo para pontuação:

|  |  |  |  |
| --- | --- | --- | --- |
|  | BAIXA | MÉDIA | ALTA |
| ALI | 7 | 10 | 15 |
| AJE | 5 | 7 | 10 |
| EE | 3 | 4 | 6 |
| SE | 4 | 5 | 7 |
| CE | 3 | 4 | 6 |

**6X EE (BAIXA) = 6\*3 = 18**

**2X CE (BAIXA) = 2\*3 = 6**

**1X SE (BAIXA) = 4\*1 = 4**

**TOTAL: 28**

**4.2 Métricas de acompanhamento**

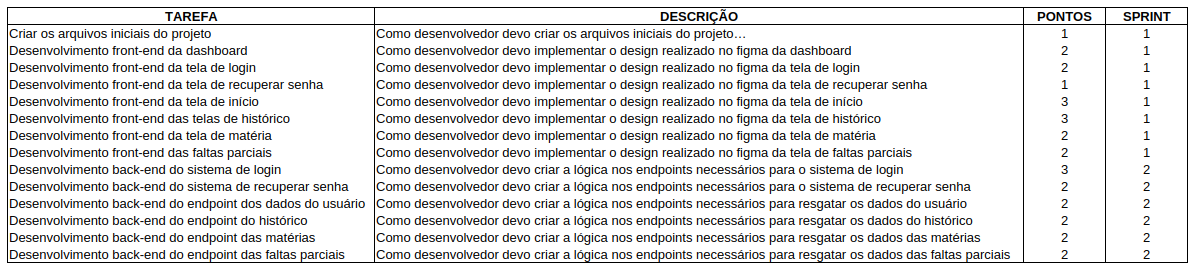
Para o acompanhamento do software é ideal que tenha um ótimo planejamento e separe nos seguintes tópicos:

* **Testes:** Liberação para testes ajuda a garantir a qualidade do software e a identificar áreas que precisam de mais testes.
* **Tempo de Resolução de Defeitos:** O tempo médio necessário para resolver defeitos ou problemas encontrados no sistema, permitindo avaliar a eficiência da equipe de desenvolvimento e a qualidade do código.
* **Taxa de Aceitação de Usuário:** É essencial termo o aceite do usuário final e ter o acompanhamento para focarmos sempre no objetivo do cliente.
* **Tempo de Resposta do Sistema:** Temos que acompanhar o tempo médio de resposta final, e verificar se está de acordo para o usuário final.
* **Satisfação do Usuário:** Fazermos pesquisas e feedbacks com o cliente para assim avaliarmos o nosso desenvolvimento no projeto e aprimorá-lo ainda mais.
* **Custo de Manutenção:** Precisamos avaliar o custo de manutenção futura para o cliente, correções de bugs, melhorias entre outros.
* **Eficiência de Desenvolvimento:** Devemos sempre ser eficientes no nosso processo para entregarmos com mais rapidez, com qualidade e menos custo.

1. **SCRUM**

**5.1 Backlog**

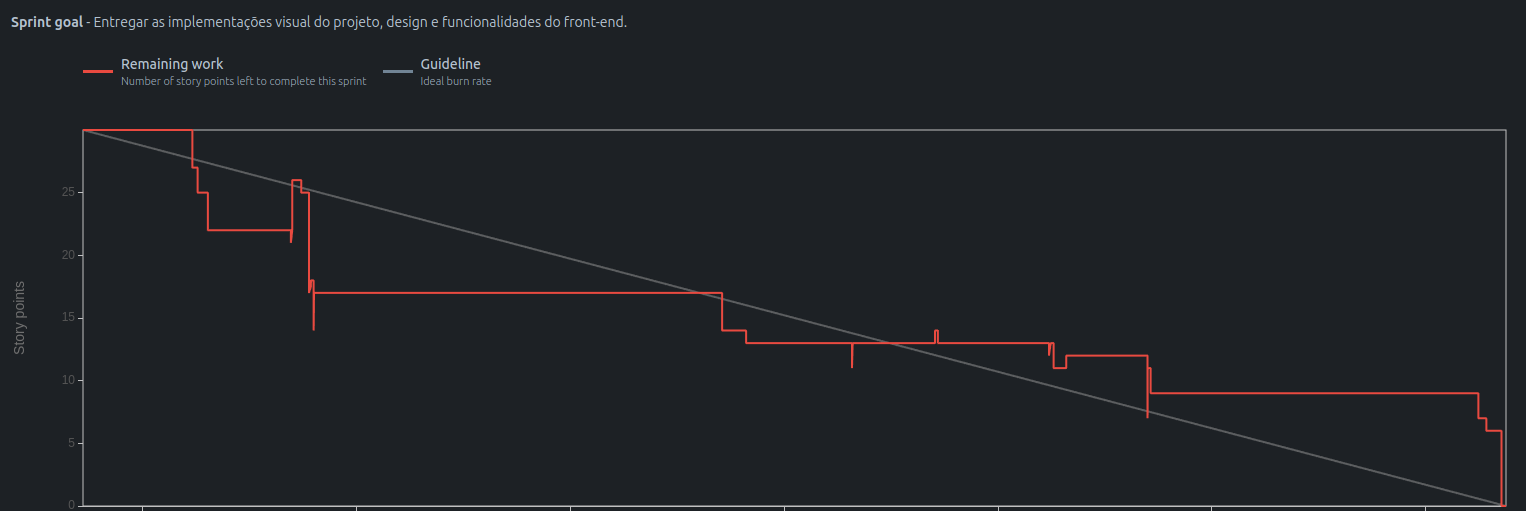
Para garantir um desenvolvimento organizado e eficiente, definimos um total de 15 tarefas que serão distribuídas entre uma equipe pequena de desenvolvedores ao longo de duas *sprints*, cada uma com a duração de uma semana. Essas tarefas abrangem tanto o desenvolvimento do *Front-end* quanto do *Back-end,* além da crucial integração entre ambos, garantindo que tudo funcione perfeitamente em conjunto. As tarefas foram planejadas de forma a equilibrar a carga de trabalho e permitir entregas incrementais, possibilitando ajustes rápidos conforme necessário.



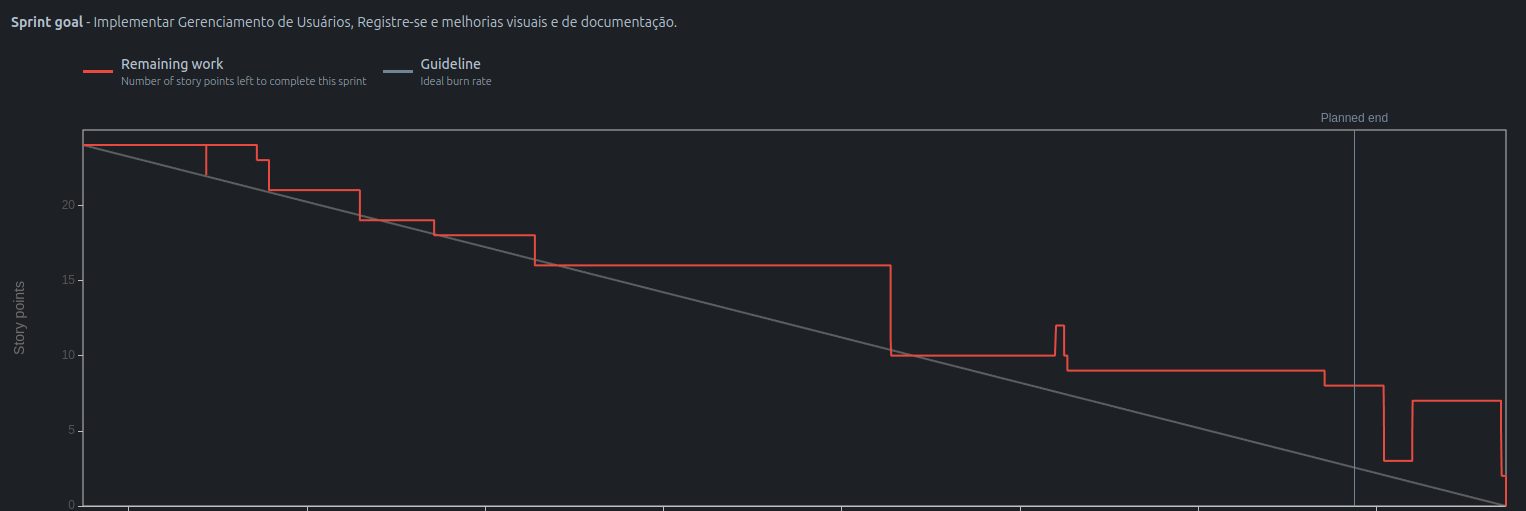
**5.2 Burndown Chart**

Para monitorar o progresso da equipe durante as duas sprints, realizamos uma simulação do desenvolvimento e geramos o *burndown chart* correspondente para cada *sprint*. Utilizamos o *Jira Agile* como a ferramenta principal para implementar o *Scrum* e criar os *burndown charts*. O *Jira Agile* nos permitiu acompanhar o avanço das tarefas de forma visual e eficiente, ajudando a identificar possíveis impedimentos e a ajustar o planejamento conforme necessário. Através do *burndown chart*, conseguimos visualizar claramente o ritmo de trabalho da equipe e garantir que estamos no caminho certo para cumprir todos os objetivos dentro do prazo estipulado.

* **SPRINT 1:**



* **SPRINT 2:**



1. **TESTES**

Em anexo junto ao trabalho deixarei a planilha de testes do Aplicativo com funcionalidades do SIGA.