CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE CAMPINAS

CURSO SUPERIOR DE TECNOLOGIA EM GESTÃO DA TECNOLOGIA DA INFORMAÇÃO

FELIPE FERNANDEZ SILVA

FELIPE FABIAN MESSIAS DE PAULA

**APLICATIVO MOBILE - SIGA ALUNOS**

  CAMPINAS/SP

2024 - 3º SEMESTRE

**RESUMO**  
  
Este trabalho apresenta uma modelagem detalhada de um aplicativo mobile para acesso dos alunos às funcionalidades do SIGA, com o objetivo de aprimorar a experiência do usuário e facilitar o dia a dia acadêmico. O aplicativo permitirá aos alunos realizar diversas atividades, como consultar notas, horários de aulas, calendário acadêmico, ementas de disciplinas, além de possibilitar a realização de matrículas e acompanhamento de suas situações acadêmicas. O projeto abrange desde a identificação de requisitos funcionais e não funcionais, até a criação de diagramas de casos de uso, propostas de interface, gestão de configuração, cálculo de métricas, planejamento e simulação de sprints, e elaboração de um plano de testes.

**1 - REQUISITOS**

**Atores:**

**Aluno**: O aluno deve ser capaz de entrar no aplicativo com suas credenciais (login e senha), consultar informações como notas, faltas, datas de provas, avisos e fazer solicitações;

**Siga:** É o sistema que interage com o usuário ao cadastrar, alterar dados e que serve como plataforma do aluno com a instituição de ensino.

**Requisitos Funcionais:**

**1.** **Realizar Matrícula**

* **Descrição:** Permitir que o aluno realize a matrícula nas disciplinas do próximo semestre.
* **Funcionalidade:** O sistema deve apresentar uma lista de disciplinas disponíveis, permitindo que o aluno selecione as desejadas e finalize a matrícula.
* **Pré-condições:** O período de matrícula deve estar aberto.
* **Pós-condições:** As disciplinas escolhidas são registradas no sistema e a matrícula é concluída com sucesso.

**2.** **Consultar Grade**

* **Descrição:** Permitir que o aluno visualize a grade de horários das disciplinas matriculadas.
* **Funcionalidade:** O sistema deve exibir a grade de horários atualizada, incluindo disciplinas, horários e salas de aula.
* **Pré-condições:** O aluno deve estar matriculado em disciplinas para o período atual.
* **Pós-condições:** A grade é exibida corretamente para o aluno.

**3.** **Consultar Histórico**

* **Descrição:** Permitir que o aluno consulte seu histórico acadêmico completo.
* **Funcionalidade:** O sistema deve mostrar todas as disciplinas cursadas pelo aluno, com suas respectivas notas, semestres e situação (aprovado, reprovado, etc.).
* **Pré-condições:** O aluno deve estar autenticado no sistema.
* **Pós-condições:** O histórico acadêmico é exibido corretamente para o aluno.

**4.** **Consultar Notas**

* **Descrição:** Permitir que o aluno consulte as notas das disciplinas do semestre atual.
* **Funcionalidade:** O sistema deve exibir as notas obtidas pelo aluno em todas as disciplinas matriculadas no semestre atual.
* **Pré-condições:** O aluno deve estar matriculado no semestre atual.
* **Pós-condições:** As notas das disciplinas do semestre atual são exibidas corretamente para o aluno.

**5.** **Consultar Faltas**

* **Descrição:** Permitir que o aluno visualize suas faltas nas disciplinas do semestre atual.
* **Funcionalidade:** O sistema deve exibir a quantidade de faltas do aluno em cada disciplina do semestre atual.
* **Pré-condições:** O aluno deve estar matriculado no semestre atual.
* **Pós-condições:** As faltas do aluno são exibidas corretamente para cada disciplina.

**6**. **Consultar Agenda Escolar**

* **Descrição:** Permitir que o aluno consulte a agenda de eventos escolares.
* **Funcionalidade:** O sistema deve exibir eventos como provas, datas de entrega de trabalhos, reuniões e outros eventos relevantes.
* **Pré-condições:** O aluno deve estar autenticado no sistema.
* **Pós-condições:** A agenda escolar é exibida corretamente para o aluno.

**7.** **Consultar Planos de Ensino**

* **Descrição:** Permitir que o aluno visualize os planos de ensino de suas disciplinas.
* **Funcionalidade:** O sistema deve disponibilizar os planos de ensino detalhados para cada disciplina, incluindo objetivos, cronograma, bibliografia, e critérios de avaliação.
* **Pré-condições:** O aluno deve estar matriculado nas disciplinas.
* **Pós-condições:** Os planos de ensino são exibidos corretamente para cada disciplina.

**8.** **Enviar Comprovantes**

* **Descrição:** Permitir que o aluno envie comprovantes de pagamento, matrícula, ou outros documentos necessários.
* **Funcionalidade:** O sistema deve permitir o upload de documentos, que serão armazenados e associados ao perfil do aluno.
* **Pré-condições:** O aluno deve estar autenticado no sistema.
* **Pós-condições:** Os comprovantes são armazenados corretamente e ficam disponíveis para consulta pela administração.

**9**. **Fazer Solicitações**

* **Descrição:** Permitir que o aluno faça solicitações diversas, como pedidos de revisão de nota, atestados de matrícula, ou declarações.
* **Funcionalidade:** O sistema deve fornecer formulários para diferentes tipos de solicitações, que serão enviados para processamento pela administração.
* **Pré-condições:** O aluno deve estar autenticado no sistema.
* **Pós-condições:** As solicitações são enviadas corretamente e ficam aguardando processamento.

**10.** **Alterar E-mail e Senha**

* **Descrição:** Permitir que o aluno altere seu e-mail de contato e senha de acesso ao sistema.
* **Funcionalidade:** O aluno deve poder acessar uma tela de configurações da conta, onde poderá atualizar seu e-mail e senha. Após a alteração, o sistema deve enviar um e-mail de confirmação para o novo endereço de e-mail.
* **Pré-condições:** O aluno deve estar autenticado no sistema.
* **Pós-condições:** O e-mail e a senha do aluno são atualizados e utilizados nas próximas tentativas de login.

**Requisitos Não Funcionais:**

**1. Desempenho:** O sistema deve garantir tempos de resposta rápidos para todas as operações, como consultas de notas, matrículas e envio de comprovantes.

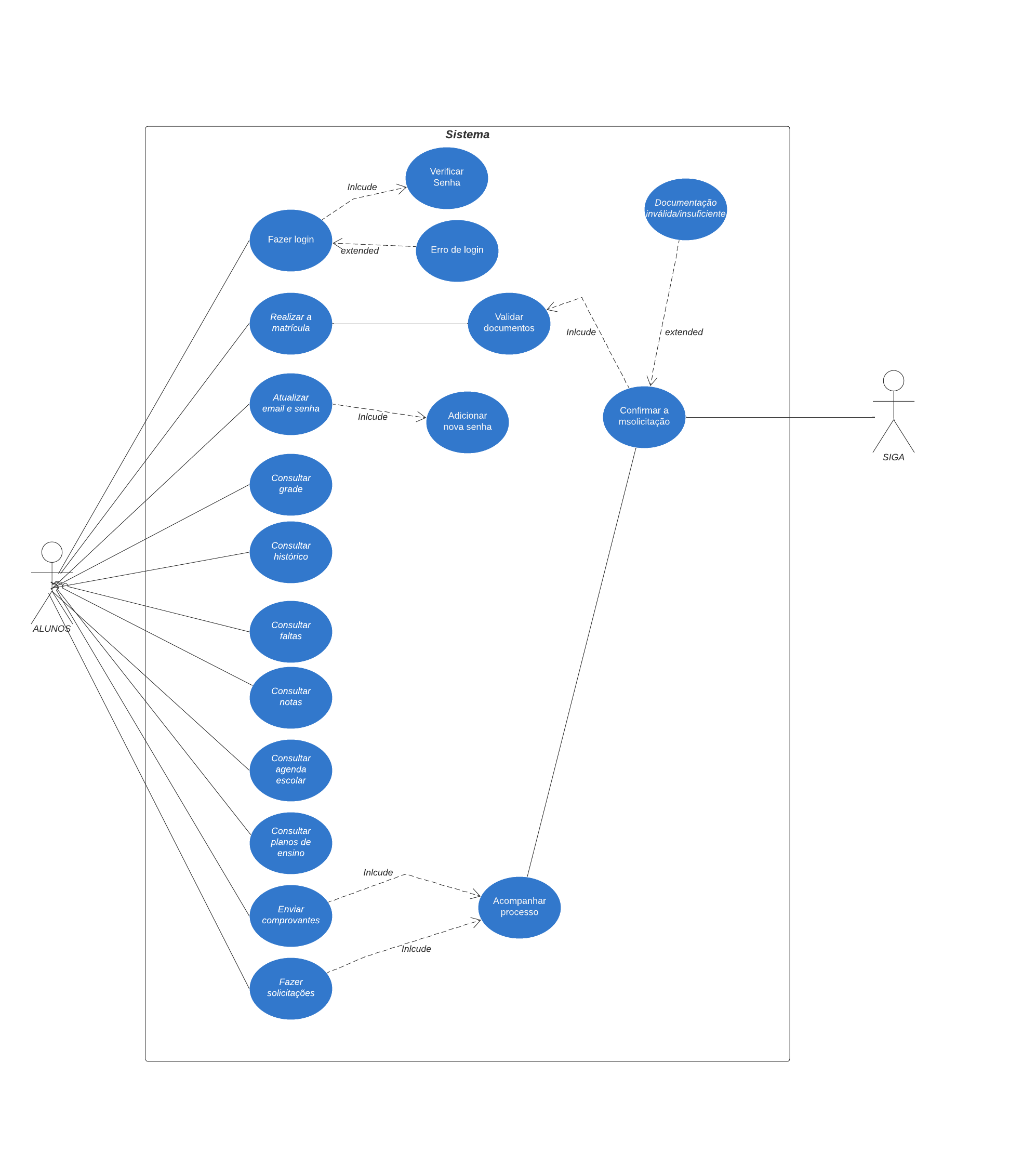
**2. Disponibilidade:** O sistema deve estar disponível para uso pelos alunos a qualquer dia e horário, exceto durante períodos de manutenção programada.

**3. Compatibilidade:** O sistema deve ser compatível com os diferentes dispositivos móveis e sistemas operacionais permitindo o acesso a partir dos diversos navegadores disponíveis

**4. Segurança:** O sistema deve garantir a segurança dos dados dos alunos e proteger contra acessos não autorizados e ataques cibernéticos.

**5. Usabilidade:** O sistema deve possuir uma interface simples e intuitiva para propiciar agilidade ao usuário.

* **Diagrama de casos de uso:**



* **Narrativa de casos de uso: Realizar Matrícula**

1. O aluno acessa o aplicativo e seleciona a opção "Realizar Matrícula".
2. O sistema exibe as disciplinas disponíveis para matrícula.
3. O aluno seleciona as disciplinas desejadas.
4. O aluno confirma a seleção.
5. O sistema verifica a disponibilidade de vagas nas disciplinas selecionadas.
6. O sistema registra a matrícula do aluno nas disciplinas confirmadas.
7. O sistema exibe uma mensagem de confirmação da matrícula.

* **Narrativa de casos de uso: Consultar Grade**

1. O aluno acessa o aplicativo e seleciona a opção "Consultar Grade".
2. O sistema recupera a grade curricular do aluno.
3. O sistema exibe a grade curricular atualizada para o aluno.

* **Narrativa de caso de uso: Consultar Histórico**

1. O aluno acessa o aplicativo e seleciona a opção "Consultar Histórico".
2. O sistema recupera o histórico acadêmico do aluno.
3. O sistema exibe o histórico acadêmico para o aluno.

* **Narrativa de caso de uso: Consultar Notas**

1. O aluno acessa o aplicativo e seleciona a opção "Consultar Notas".
2. O sistema recupera as notas das disciplinas cursadas pelo aluno.
3. O sistema exibe as notas para o aluno.

* **Narrativa de caso de uso: Consultar Agenda escolar**

1. O aluno acessa o aplicativo e seleciona a opção "Consultar Agenda Escolar".
2. O sistema recupera os eventos e compromissos da agenda escolar do aluno.
3. O sistema exibe a agenda escolar para o aluno.

* **Narrativa de caso de uso: Consultar Planos de ensino**

1. O aluno acessa o aplicativo e seleciona a opção "Consultar Agenda Escolar".
2. O sistema recupera os eventos e compromissos da agenda escolar do aluno
3. O sistema exibe a agenda escolar para o aluno

* **Narrativa de caso de uso: Enviar Comprovantes**  
  O aluno acessa o aplicativo e seleciona a opção "Enviar Comprovantes".

1. O aluno escolhe o tipo de comprovante a ser enviado.
2. O aluno faz o upload do arquivo do comprovante.
3. O aluno confirma o envio.
4. O sistema armazena o comprovante e exibe uma mensagem de confirmação.

* **Narrativa de caso de uso: Fazer Solicitações**  
  O aluno acessa o aplicativo e seleciona a opção "Fazer Solicitações".

1. O aluno escolhe o tipo de solicitação a ser feita (Exame, Documentos, Biblioteca ou Exercícios Domiciliares).
2. O aluno preenche os detalhes da solicitação.
3. O aluno confirma o envio da solicitação.
4. O sistema registra a solicitação e envia para análise.
5. O sistema exibe uma mensagem de confirmação.

* **Narrativa de caso de uso: Alterar e-mail e senha**

1. O aluno acessa o aplicativo e seleciona a opção "Alterar E-mail e Senha".
2. O aluno escolhe se deseja alterar o e-mail ou a senha.
3. O aluno insere o novo e-mail ou a nova senha.
4. O aluno confirma a alteração.
5. O sistema valida e atualiza o e-mail ou a senha.
6. O sistema exibe uma mensagem de confirmação.

**2 - INTERFACE**

**Recomendações de Nielsen**

1. Visibilidade do status do sistema: o aplicativo deve sempre manter os alunos; informados sobre o que está acontecendo, através de feedback adequado em tempo razoável;
2. Compatibilidade entre o sistema e o mundo real: utilizar a linguagem dos alunos, com palavras, frases e conceitos familiares ao usuário, ao invés de termos técnicos
3. Controle e liberdade do usuário: facilitar a navegação com botões de voltar e sair facilmente acessíveis, além de permitir desfazer ações;
4. Consistência e padrões: manter consistência na navegação, layout e design em todas as telas;
5. Prevenção de erros: projetar o sistema para prevenir a ocorrência de erros;
6. Reconhecimento em vez de lembrança: reduzir a carga cognitiva tornando objetos, ações e opções visíveis;
7. Flexibilidade e eficiência de uso: permitir atalhos e opções de personalização para usuários experientes;
8. Estética e design minimalista: evitar exibir informações irrelevantes ou raramente necessárias;
9. Ajuda e documentação: disponibilizar ajuda e documentação facilmente acessível e focada na tarefa dos usuários.

**Boas Práticas de Projeto de Interface**

1. Layout responsivo: garantir que a interface se adapte a diferentes tamanhos de tela e dispositivos;
2. Tipografia legível: utilizar fontes claras e tamanhos adequados para facilitar a leitura;
3. Navegação intuitiva: menu de fácil acesso e claramente definido;
4. Cores e Contrates: utilizar cores de alto contraste para garantir a legibilidade e acessibilidade;
5. Feedback imediato: fornecer feedback instantâneo para as ações do usuário (e.g., mensagens de sucesso ou erro);
6. Carga rápida: otimizar o desempenho para garantir tempos de carregamento rápidos;
7. Acessibilidade: seguir diretrizes de acessibilidade para garantir que todos os alunos possam usar o aplicativo;
8. Coerência visual: manter um design visualmente coerente em todas as telas;
9. Ícones claros: utilizar ícones reconhecíveis para ações comuns;
10. Testes de usabilidade: realizar testes com usuários reais com intuito de identificar e corrigir problemas de usabilidade.

**Proposta de Interface (Telas e Navegação)**

**1 – Tela de Login**

Elementos:

* Campo de texto para “Email”;
* Campo de texto para “Senha”;
* Botão “Salvar Credenciais”;
* Botão “Entrar”.

**2 – Tela Principal (Dashboard)**

Elementos:

* Campo de texto para “Pesquisar”;
* Botão “Dados do Aluno”;
* Botão “Planos de Ensino;
* Botão “Atestados e Declarações”;
* Botão “Consultas”.

**3 – Tela de Disciplinas**

Elementos:

* Menu com a lista de Disciplinas;
* Dashboard com informações referentes a notas de avaliações, notas de trabalhos, média atual, quantidade de aulas totais, quantidade de presenças, quantidade de faltas e porcentagem de presenças da disciplina selecionada;
* Botão “Voltar” para retornar a tela principal.

**4 – Tela de Eventos Futuros**

Elementos:

* Botão Vestibular 2º semestre 2024;
* Botão “XV congresso de logística”;
* Botão “Desbravando o Universo da Modelagem 3D”;
* Botão “Feira da Empregabilidade”;
* Botão “Voltar” para retornar a tela principal.

Para a criação dos Wire frames, pode-se utilizar ferramentas como o Figma, Balsamiq ou o Adobe XD. Essas ferramentas permitem criar protótipos interativos que ajudam na visualização e navegação das telas conforme descrito acima. Este tipo de abordagem garante uma interface bem planejada, centrada no usuário, que atende às necessidades dos alunos ao acessar as funcionalidades do SIGA através do aplicativo móvel.

**3 - GESTÃO DE CONFIGURAÇÃO DE SOFTWARE**

**Proposta de Organização de Configuração**

**1 - Identificação de Configuração**

A Identificação de Configuração define quais itens de software compõem a configuração. Nesta proposta, os seguintes componentes são identificados:

* Código Fonte
* Documentos do Sistema
* Tabelas de Parâmetros
* Estruturas de Arquivos

**2 - Controle de Configuração**

O Controle de Configuração garante que os itens de configuração estejam de acordo com os requisitos do projeto e sejam protegidos contra alterações não autorizadas. Os passos no processo de alteração que afetam uma configuração incluem:

* Configuração Básica (Baseline): Estabelecimento de uma baseline, onde itens de configuração são considerados conformes aos requisitos do projeto.
* Controle de Alterações: Após o estabelecimento da baseline, todas as alterações nos itens de configuração devem ser formalmente controladas para garantir a integridade do produto.

**3 - Auditoria de Configuração**

A Auditoria de Configuração verifica as diferenças entre versões e assegura que o produto está de acordo com os requisitos do projeto. Os principais pontos incluem:

* Auditoria de Produto: Realizada por um grupo independente, garantindo que o produto atende aos requisitos especificados e contratuais.
* Auditoria de Alterações: Todas as alterações devem ser auditadas para garantir que o status do software seja mantido ao longo do ciclo de vida.

**4 - Administração de Estados**

A Administração de Estados monitora as modificações feitas e seus autores. Inclui:

* Rastreamento de Modificações: Identificação do autor, tipo de modificação e momento da alteração, permitindo uma gestão eficiente do desenvolvimento do software.

**Criação de Projeto e Inclusão de Documentos**

* O projeto incluirá os seguintes tipos de documentos:
* Suporte ao Usuário Final: Guia de usuário, notas de versão, sistemas de ajuda online, programas de treinamento e procedimentos operacionais.
* Suporte de Marketing: Publicidade focada no produto, incluindo fotos, vídeos explicativos, apresentações e materiais promocionais.
* Suporte ao Desenvolvimento: Especificações funcionais e técnicas, guias de desenvolvimento, procedimentos e ferramentas para desenvolvedores.
* Suporte à Organização: Informações sobre a empresa, estrutura, procedimentos, fluxos de trabalho e políticas internas.

**Ferramentas Utilizadas**

* As ferramentas utilizadas para a gestão de configuração incluem:
* Microsoft Word: Para criação e edição de documentos.
* Microsoft SharePoint: Para armazenamento e compartilhamento de documentos.
* Microsoft Teams: Para comunicação e colaboração entre os membros da equipe.
* Adobe Photoshop: Para criação e edição de materiais de marketing visuais.

**4 - MÉTRICAS**

Para garantir um desenvolvimento eficiente e a entrega de um produto de alta qualidade, é essencial definir métricas claras e precisas. A seguir, apresentamos as métricas que serão utilizadas no desenvolvimento do Sistema SIGA, incluindo o cálculo do número de pontos de função estimado.

**1 - Métricas de Produto**

Essas métricas medem características do produto, como tamanho, complexidade e qualidade.

**Tamanho do Software**

* Pontos de Função (PF): Método para medir o tamanho funcional do software, baseado na funcionalidade fornecida ao usuário final. Abaixo está o cálculo detalhado.

**Qualidade do Software**

* Densidade de Defeitos: Número de defeitos encontrados por mil linhas de código ou por pontos de função.
* Taxa de Defeitos Pós-Entrega: Número de defeitos encontrados após a entrega do software, por mil linhas de código ou por pontos de função.

**Complexidade do Software**

* Complexidade Ciclomática: Mede a complexidade do código, calculada a partir do grafo de controle de fluxo

**2 - Métricas de Processo**

Essas métricas avaliam o processo de desenvolvimento, com foco na eficiência e eficácia.

**Desempenho da Equipe**

* Velocidade: Número de pontos de função concluídos por iteração.
* Produtividade: Número de pontos de função entregues por pessoa por mês.

**Aderência ao Cronograma**

* Desvio do Cronograma: Diferença entre o tempo planejado e o tempo real gasto.
* Cumprimento de Prazo: Percentual de entregas realizadas dentro do prazo estabelecido.

**Eficiência do Processo**

* Taxa de Retrabalho: Percentual de esforço gasto em correções e melhorias.
* Tempo Médio para Reparação (MTTR): Tempo médio gasto para corrigir defeitos

**3 - Métricas de Projeto**

Essas métricas ajudam a monitorar a saúde geral do projeto.

**Riscos**

* Número de Riscos Identificados: Total de riscos identificados durante o projeto.
* Taxa de Mitigação de Riscos: Percentual de riscos mitigados com sucesso.

**Recursos**

* Uso de Recursos: Medição do uso eficiente dos recursos disponíveis, como tempo, equipe e orçamento.
* Custo por Ponto de Função: Custo total dividido pelo número de pontos de função.

**4 – Tabela de Contagem e Cálculo dos Pontos de Função**

**Entradas Externas (EE):**

* Realizar Matrícula: 4 (Complexidade Média)
* Enviar Comprovantes: 3 (Complexidade Baixa)
* Fazer Solicitações: 3 (Complexidade Baixa)
* Alterar E-mail e Senha: 3 (Complexidade Baixa)

**Saídas Externas (SE):**

* Consultar Notas: 4 (Complexidade Média)
* Consultar Faltas: 4 (Complexidade Média)
* Consultar Histórico: 5 (Complexidade Alta)
* Consultar Agenda Escolar: 3 (Complexidade Baixa)
* Consultar Planos de Ensino: 4 (Complexidade Média)

**Consultas Externas (CE):**

* Consultar Grade: 4 (Complexidade Média)

**Arquivos Lógicos Internos (ALI):**

* Cadastro de Alunos: 7 (Complexidade Alta)
* Cadastro de Disciplinas: 5 (Complexidade Alta)
* Cadastro de Notas: 5 (Complexidade Alta)

**Arquivos de Interface Externa (AIE):**

* Interface com Sistema de PGTO.: 5 (Complexidade Alta)

**Tabela de Fatores de Ajuste**

1 - Comunicação de Dados: 2

2 - Processamento Distribuído: 3

3 - Performance: 2

4 - Utilização do Usuário Final: 3

5 - Complexidade de Transações: 3

6 - Reusabilidade: 3

7 - Facilidade de Implementação: 2

8 - Facilidade de Operação: 2

9 - Portabilidade: 2

10 - Facilidade de Mudança: 3

11 - Facilidade de Instalação: 2

12 - Facilidade de Uso: 3

13 - Documentação: 3

**Cálculo do Número de Pontos de Função**

**Total não ajustado:**

EE = 13

SE = 20

CE = 4

ALI = 17

AIE = 5

Total: 13 + 20 + 4 + 17 + 5 = 59

**Fator de Ajuste:**

Soma dos fatores: 35

Fator de Ajuste: 0.65 + (0.01 \* 35) = 1.00

Total ajustado:

Pontos de Função = 59 \* 1.00 = 59 PF

Com o cálculo detalhado de pontos de função e a definição das métricas de produto, processo e projeto, o desenvolvimento do Sistema SIGA está bem fundamentado. Essas métricas não apenas permitirão monitorar e controlar o progresso do desenvolvimento, mas também assegurarão a qualidade e a eficiência do produto.

**5 -** **DEFINIÇÃO DO BACKLOG, PRIMEIRA SPRINT E BURNDOWN CHART**

**Definição do Backlog do Sistema**

**1 - Funcionalidades Principais:**

* Realizar Matrícula;
* Consultar Grade;
* Consultar Histórico;
* Consultar Notas;
* Consultar Faltas;
* Consultar Agenda Escolar;
* Consultar Planos de Ensino;
* Enviar Comprovantes;
* Fazer Solicitações;
* Alterar E-mail e Senha.

**2 - Requisitos Não Funcionais:**

* Desempenho;
* Disponibilidade;
* Compatibilidade;
* Segurança;
* Usabilidade.

**3 - Backlog Detalhado**

**Realizar Matrícula**

* Descrição: Como aluno, quero poder me matricular nas disciplinas do próximo semestre.
* Tarefas:
  + Criar interface de seleção de disciplinas
  + Implementar lógica de matrícula
  + Validar período de matrícula aberto
  + Confirmar matrícula ao aluno
  + Estimativa: 8 pontos

**Consultar Grade**

* Descrição: Como aluno, quero ver minha grade de horários para o semestre atual.
* Tarefas:
  + Criar interface de visualização de grade
  + Recuperar dados da grade do sistema
  + Exibir grade para o aluno
  + Estimativa: 5 pontos

**Consultar Histórico**

* Descrição: Como aluno, quero consultar meu histórico acadêmico completo.
* Tarefas:
  + Criar interface de histórico acadêmico
  + Recuperar dados do histórico
  + Exibir histórico para o aluno
  + Estimativa: 5 pontos

**Consultar Notas**

* Descrição: Como aluno, quero consultar minhas notas do semestre atual.
* Tarefas:
  + Criar interface de notas
  + Recuperar dados das notas
  + Exibir notas para o aluno
  + Estimativa: 5 pontos

**Consultar Faltas**

* Descrição: Como aluno, quero ver minhas faltas nas disciplinas do semestre atual.
* Tarefas:
  + Criar interface de faltas
  + Recuperar dados das faltas
  + Exibir faltas para o aluno
  + Estimativa: 5 pontos

**Consultar Agenda Escolar**

* Descrição: Como aluno, quero consultar a agenda de eventos escolares.
* Tarefas:
  + Criar interface de agenda escolar
  + Recuperar dados da agenda
  + Exibir agenda para o aluno
  + Estimativa: 5 pontos

**Consultar Planos de Ensino**

* Descrição: Como aluno, quero ver os planos de ensino das minhas disciplinas.
* Tarefas:
  + Criar interface de planos de ensino
  + Recuperar dados dos planos de ensino
  + Exibir planos de ensino para o aluno
  + Estimativa: 5 pontos

**Enviar Comprovantes**

* Descrição: Como aluno, quero enviar comprovantes de pagamento, matrícula, ou outros documentos.
* Tarefas:
  + Criar interface de envio de comprovantes
  + Implementar funcionalidade de upload
  + Armazenar comprovantes no sistema
  + Estimativa: 5 pontos

**Fazer Solicitações**

* Descrição: Como aluno, quero fazer solicitações como revisão de nota, atestados ou declarações.
* Tarefas:
  + Criar interface de solicitações
  + Implementar formulários de solicitação
  + Enviar solicitações para processamento
  + Estimativa: 5 pontos

**Alterar E-mail e Senha**

* Descrição: Como aluno, quero alterar meu e-mail de contato e senha de acesso.
* Tarefas:
  + Criar interface de configurações da conta
  + Implementar funcionalidade de alteração de e-mail e senha
  + Validar e confirmar alterações
  + Estimativa: 5 pontos

**Primeira Sprint (2 semanas)**

**Objetivo da Sprint**

Implementar as funcionalidades básicas de acesso e consulta de informações acadêmicas, focando nas funcionalidades de login, matrícula, consulta de notas, e consulta de grade.

**Ações Selecionadas:**

* Realizar Matrícula (8 pontos)
* Consultar Grade (5 pontos)
* Consultar Notas (5 pontos)
* Alterar E-mail e Senha (5 pontos)

Total de Pontos na Sprint: 23 pontos

**Simulação da Execução da Primeira Sprint**

Dia 1-2:

* Planejamento da Sprint
* Tarefas de configuração do ambiente de desenvolvimento
* Design inicial das interfaces

Dia 3-5:

* Desenvolvimento da funcionalidade de "Realizar Matrícula"
* Testes unitários e de integração para matrícula

Dia 6-8:

* Desenvolvimento da funcionalidade de "Consultar Grade"
* Testes unitários e de integração para consulta de grade

Dia 9-10:

* Desenvolvimento da funcionalidade de "Consultar Notas"
* Testes unitários e de integração para consulta de notas

Dia 11-13:

* Desenvolvimento da funcionalidade de "Alterar E-mail e Senha"
* Testes unitários e de integração para alteração de e-mail e senha

Dia 14:

* Testes de integração final
* Revisão da Sprint
* Preparação para a próxima Sprint

**Burndown Chart**

O burndown chart será atualizado diariamente para acompanhar o progresso da Sprint. Um exemplo de burndown chart é mostrado abaixo:

Dia | Pontos Restantes  
-----|-----------------  
1 | 23  
2 | 22  
3 | 20  
4 | 18  
5 | 15  
6 | 14  
7 | 12  
8 | 10  
9 | 8  
10 | 6  
11 | 4  
12 | 3  
13 | 1  
14 | 0

**Ferramentas Utilizadas**

Para a gestão da Sprint e monitoramento do progresso, utilizaremos as seguintes ferramentas:

* Jira Agile: Para a gestão do backlog, planejamento da Sprint e acompanhamento das tarefas.
* Trello: Para visualização de tarefas e progresso da Sprint.
* GitHub: Para controle de versão do código e colaboração entre os desenvolvedores.
* Monday: Para monitoramento das tarefas e comunicação entre a equipe.

A definição detalhada do backlog, planejamento da primeira Sprint e simulação da execução fornecem uma base sólida para o desenvolvimento do Sistema SIGA. A utilização de ferramentas de gestão ágil como Jira, Trello e Monday assegura uma gestão eficiente do projeto, enquanto o burndown chart facilita o acompanhamento do progresso da Sprint.

**6 - TESTES PARA O SISTEMA SIGA**

**Objetivo**

O plano de testes visa garantir que todas as funcionalidades do Sistema SIGA estejam funcionando corretamente e de acordo com os requisitos especificados. Ele cobre diversos tipos de testes, incluindo testes funcionais, de integração, de desempenho, de segurança e de usabilidade.

**Escopo**

O escopo inclui todas as funcionalidades descritas nos requisitos funcionais, bem como os requisitos não funcionais relacionados ao desempenho, disponibilidade, compatibilidade, segurança e usabilidade.

**Tipos de Teste**

* **Testes Funcionais:**
  + Testar todas as funcionalidades descritas nos requisitos funcionais.
  + Verificar a correção e completude dos dados exibidos.
* **Testes de Integração:**
  + Testar a interação entre diferentes módulos do sistema.
  + Garantir que os dados fluam corretamente entre os módulos.
* **Testes de Desempenho:**
  + Verificar tempos de resposta e desempenho sob diferentes cargas de usuários.
  + Testar a escalabilidade do sistema.
* **Testes de Segurança:**
  + Garantir que os dados dos alunos estão protegidos.
  + Testar a resistência do sistema contra-ataques cibernéticos.
* **Testes de Usabilidade:**
  + Avaliar a facilidade de uso do sistema.
  + Verificar a consistência da interface e a satisfação dos usuários.

**Estrutura do Plano de Testes**

* Introdução:
  + Visão geral do sistema SIGA.
  + Objetivo dos testes.
* Escopo:
  + Funcionalidades e módulos cobertos pelos testes.
* Estratégia de Testes:
  + Tipos de testes a serem realizados.
  + Ferramentas e ambientes de teste.
* Recursos de Teste:
  + Equipe de teste.
  + Ferramentas de gerenciamento e execução de testes (e.g., Jira, Trello).
* Cronograma de Testes:
  + Datas importantes e marcos do teste.
* Critérios de Aceitação:
  + Definição de sucesso e critérios para aceitação dos testes.

**REFERÊNCIAS BIBLIOGRÁFICAS**

<https://siga.cps.sp.gov.br/ALUNO/login.aspx>https://siga.cps.sp.gov.br

<https://www.figma.com><https://www.lucidchart.com>