

GamaTrack

VISÃO DO PRODUTO E DO PROJETO

Versão 2.0

Histórico de Revisão

Data	Versão	Descrição	Autor
02/10/2023	1.0	Início do Documento Visão.	Nicollas Gabriel Samuel Ribeiro Eric Rabelo Isaque Coleman Rodrigo Braz
09/11/2023	2.0	Aplicando alterações ao documento de acordo com a correção do professor Ricardo Ajax.	Nicollas Gabriel Samuel Ribeiro Eric Rabelo Isaque Coleman Rodrigo Braz

Sumário

1.1. Problema	3
1.2. Declaração de Posição do Produto	3
1.3. Objetivos do Produto	4
1.4. Tecnologias a Serem Utilizadas	4
2. <i>VISÃO GERAL DO PROJETO</i>	5
2.1. Ciclo de vida do projeto de desenvolvimento de software.....	5
2.2. Organização do Projeto	5
2.3. Planejamento das Fases e/ou Iterações do Projeto	6
2.4. Matriz de Comunicação.....	6
2.5. Gerenciamento de Riscos	6
2.6. Critérios de Replanejamento	7
3. <i>Processo de desenvolvimento de Softwa</i>	8
4. <i>Detalhamento de atividades do projeto</i>	7
4.1. Sprint 1.....	7
4.2. Sprint 2.....	8
4.3. Sprint 3.....	8
4.4. Sprint 4.....	8
5. <i>LIÇÕES APRENDIDAS</i>	8
5.1. Unidade 1	9
5.2. Unidade 2	9
5.3. Unidade 3	9
5.4. Unidade 4	9
6. <i>PRÓXIMOS PASSOS</i>	9
7. <i>REFERÊNCIAS BIBLIOGRÁFICAS</i>	9

1. VISÃO DO PRODUTO E PROJETO

1.1. Problema

- **Contexto:** Atualmente, os alunos da FGA enfrentam desafios ao fornecer e receber feedback sobre os professores de forma eficiente e anônima.
- **Problema:** Como proporcionar aos estudantes da FGA um sistema eficaz que lhes permita avaliar e acessar avaliações de outros alunos sobre os professores, a fim de facilitar a escolha de suas disciplinas e professores de maneira informada e alinhada com suas preferências acadêmicas?

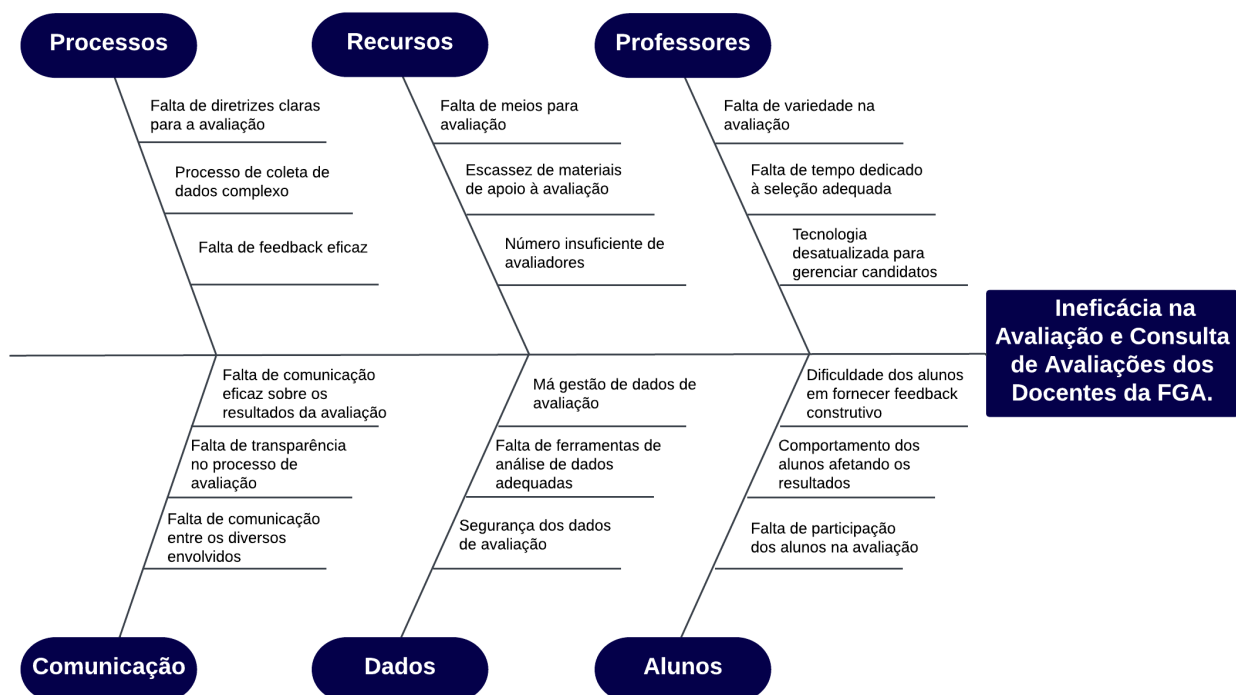


Imagem 1 - Diagrama de Ishikawa.

- **Solução de Software Proposta:** Nosso software representa uma aplicação web que oferece uma abordagem prática e em tempo real para pesquisar e avaliar os professores da FGA. A proposta nasce em um contexto de ineficácia da avaliação e consulta de outras avaliações dos docentes da FGA, como pode ser contemplado no diagrama de ishikawa acima. O propósito central desta ferramenta é criar um ambiente seguro e anônimo onde estudantes, tanto novos quanto veteranos, possam compartilhar suas opiniões sobre os professores e as disciplinas ministradas na Faculdade do Gama (FGA).

1.2. Declaração de Posição do Produto

Tabela 1 - Visão do Produto

Para:	Direcionado a todos os discentes das graduações da FGA (Faculdade do Gama). No entanto, sua validação será conduzida por um grupo de estudantes que não está matriculado na disciplina de 'Métodos de Desenvolvimento de Software'.
Necessidade:	Ineficácia na avaliação e a consulta de avaliações dos docentes por discentes da FGA.
O (nome do produto):	GamaTrack.
Que:	O software visa auxiliar os discentes a selecionar os docentes que mais se adequam às suas preferências, permitindo-lhes avaliar os professores e consultar avaliações de outros alunos. Esse processo visa possibilitar aos estudantes uma escolha mais informada, contribuindo para que possam aproveitar ao máximo as disciplinas acadêmicas.
Ao contrário:	Das atuais abordagens para avaliação e pré-seleção de docentes por discentes para o curso de uma disciplina são limitadas, que geralmente envolvem grupos em redes sociais, onde a obtenção de informações é desafiadora e suscetível a comentários de qualquer pessoa, sem restrições.
Nosso produto:	Em contraste, oferece uma solução mais eficiente e confiável, centralizando todas as avaliações e informações dos docentes da FGA em uma plataforma dedicada, garantindo um acesso mais prático e confiável a tal informação.

1.3. Objetivos do Produto

Desenvolver uma plataforma de avaliação de professores universitários que permita aos estudantes da FGA avaliar de maneira prática e anônima os professores e suas respectivas disciplinas. O principal foco é fornecer aos estudantes informações confiáveis para auxiliá-los na seleção de professores e disciplinas que melhor atendam às suas necessidades acadêmicas.

1.4. Tecnologias a Serem Utilizadas

Ferramentas de Gerenciamento de Projeto:

- **GitHub:** Utilizado para gerenciamento de código-fonte, controle de versão e colaboração entre a equipe.
- **Trello:** Usado para organizar tarefas, criar listas de afazeres e acompanhar o progresso do projeto.
- **Teams:** Uma plataforma de comunicação e colaboração que auxilia na comunicação da equipe e reuniões virtuais.

Frontend:

- **React.js:** Uma biblioteca JavaScript popular para a criação de interfaces de usuário interativas e responsivas.
- **Figma:** Uma ferramenta de design de interface de usuário (UI) que permite criar protótipos, designs e colaborar na criação da interface do usuário do seu aplicativo web.

Backend:

- **Node.js:** Uma plataforma de tempo de execução JavaScript que permite o desenvolvimento do lado do servidor.
- **Express.js:** Um framework web Node.js que simplifica o desenvolvimento de aplicativos web e APIs.
- **MongoDB:** Um sistema de gerenciamento de banco de dados NoSQL, que é escalável e adequado para armazenar dados flexíveis e não estruturados.

2. VISÃO GERAL DO PROJETO

2.1. Ciclo de vida do projeto de desenvolvimento de software

- **Metodologia:** Visando a flexibilidade, a praticidade para colaboração e a otimização do tempo disponível, a abordagem filosófica será o uso de uma metodologia ágil, ideal para desenvolver este projeto de forma eficiente e eficaz, considerando a dinamicidade necessária para alinhar objetivos e reavaliar metas ao longo da matéria. Para tanto, o grupo optou por seguir os princípios do Scrum. De acordo com Audy (2015), o SCRUM é um framework ágil que enfatiza a colaboração, a adaptação a mudanças e a entrega de valor contínuo ao cliente.
- **Processo:** O processo de desenvolvimento de software adota uma abordagem híbrida, combinando o framework Scrum para a gestão do projeto com práticas técnicas do Extreme Programming (XP) para garantir a qualidade técnica do software (PMI, 2017). Iniciamos com a fase de iniciação, onde definimos o escopo do projeto e identificamos as partes interessadas. Durante essa fase, também criamos o Product Backlog, que lista os requisitos iniciais. À medida que avançamos para a fase de Planejamento da Sprint, selecionamos um conjunto de itens do Product Backlog para a próxima Sprint e definimos os objetivos da Sprint. Ao mesmo tempo, as práticas técnicas do XP, como pair programming e integração contínua, são adotadas durante o desenvolvimento. A equipe trabalha colaborativamente para produzir código de alta qualidade, enquanto a integração contínua garante que o software seja testado regularmente para identificar problemas precocemente. Ao final de cada Sprint, realizamos a Revisão da Sprint, onde o cliente fornece feedback sobre o trabalho concluído.
- **Procedimento:** O ciclo de vida ágil, escolhido pela equipe, envolve um conjunto de procedimentos para o desenvolvimento de projetos de forma iterativa e colaborativa. Ele inclui planejamento, priorização, desenvolvimento, revisão, feedback do cliente, adaptação, monitoramento, controle e encerramento. O procedimento central deste projeto envolverá as seguintes etapas: cadastro de alunos, armazenamento de dados relevantes e interação com os usuários. Essas ações serão realizadas ao longo das iterações do Scrum para garantir a entrega contínua de valor aos envolvidos.
- **Métodos:** Nossas atividades incluirão a realização de reuniões semanais para manter a equipe alinhada e promover a comunicação eficaz. Além disso, utilizaremos o planejamento de sprints para definir metas claras e prioridades para cada iteração do projeto. Para a organização eficaz das atividades da equipe, implementaremos um quadro Kanban, permitindo o acompanhamento visual do fluxo de trabalho e facilitando a gestão das tarefas em andamento. Além disso, para uma gestão mais eficiente e transparente das tarefas, todas as atividades serão listadas em formato de "issues" no repositório da equipe Pollux no GitHub, onde poderemos monitorar o progresso, atribuir responsabilidades e colaborar de forma mais eficaz. Essas práticas e métodos combinados contribuirão para a melhoria da organização e do gerenciamento de tarefas da equipe de desenvolvimento, promovendo uma abordagem ágil e eficiente em nosso processo de desenvolvimento de software.

- **Ferramentas:** Para dar suporte ao desenvolvimento e gestão do projeto, serão utilizadas as seguintes ferramentas: ferramentas de desenvolvimento de código — Visual Studio Code com a extensão Live Share, a fim de possibilitar a prática de pair programming à distância sem grandes complicações e Insomnia, para realização dos testes de bancos de dados; ferramentas de organização de projeto (Trello e GitHub) e ferramentas de comunicação síncronas e assíncronas (Discord, Teams e WhatsApp).

2.2. Organização do Projeto

Tabela 2 – Tabela de Organização

<i>Papel</i>	<i>Atribuições</i>	<i>Responsável</i>	<i>Participantes</i>
Desenvolvedor	Codificação do produto, colaboração em equipe, aplicação de práticas técnicas e compromisso com os objetivos do produto.	<i>Todos os integrantes</i>	Nicollas Gabriel Samuel Ribeiro Eric Rabelo Isaque Colem Rodrigo Braz
Dono do Produto	Atualizar o escopo do produto, organizar o escopo das sprints, validar as entregas	<i>Todos os integrantes</i>	Grupo seletivo de estudantes da FGA não matriculados na disciplina.
Cliente	Fornecimento de requisitos e validação da aplicação.	<i>Todos os integrantes</i>	Estudantes da FGA.

2.3. Planejamento das Fases e/ou Iterações do Projeto

Tabela 3 – Planejamento e Sprint

<i>Sprint</i>	<i>Produto (Entrega)</i>	<i>Data Início</i>	<i>Data Fim</i>	<i>Entregável(eis)</i>	<i>Responsáveis</i>	<i>% conclusão</i>
Sprint 0	<ul style="list-style-type: none"> Definição do produto. 	05/09/2023	12/09/2023	Escolha do tema e definição do escopo do projeto.	Todos	100%
Sprint 1	<ul style="list-style-type: none"> Definição de tecnologias e treinamento das equipes. 	12/09/2023	19/09/2023	Linguagens e frameworks utilizados no projeto identificados. Conhecimento básico da equipe nas tecnologias.	Todos	100%
Sprint 2	<ul style="list-style-type: none"> Protótipo de telas no Figma. 	19/09/2023	25/09/2023	Todas as telas do software modeladas na equipe do Figma.	Todos	100%
Sprint 3	<ul style="list-style-type: none"> Codificação das telas Login e Cadastro. Criação do banco de dados. 	26/09/2023	03/10/2023	Estrutura das telas de login e cadastro.	Todos	100%
Sprint 4	<ul style="list-style-type: none"> CRUD do usuário. Alimentar base de dados com professores. Lógica de autenticação pela parte do usuário. Iniciar documentação no MkDocs 	03/10/2023	10/10/2023	Sistema que permite a realização das operações CRUD (Criar, Ler, Atualizar, Deletar) para usuários. Base de dados atualizada e populada com informações dos professores.	Todos	100%
Sprint 5	<ul style="list-style-type: none"> Lógica de registro do usuário pela parte do cliente Lógica de cadastro pela parte do servidor Desenvolver estrutura da página inicial Integração do Front e Back no login e cadastro Importar banco de dados no backend 	10/10/2023	17/10/2023	Implementação do sistema CRUD de usuários (US-01), a base de dados alimentada com informações de professores (US-05), a lógica de autenticação do usuário (US-02), e o início da documentação no MkDocs.	Todos	100%

Sprint 6	<ul style="list-style-type: none"> • Otimizar conexão com banco de dados • Desenvolver componente de barra de pesquisa para busca de professores • Melhorar usabilidade de login e cadastro para usuário • Implementar API para busca de professores 	17/10/2023	24/10/2023	Otimização da conexão com o banco de dados , criação do componente de barra de pesquisa para buscar professores (US-01), aprimoramento da usabilidade no processo de login e cadastro para os usuários (US-01), e a implementação da API de busca de professores (US-05).	Todos	75%
Sprint 7	<ul style="list-style-type: none"> • Desenvolver componente de card dos professores • Realizar novo webscrapping para coleta de dados dos professores do sigaa • Implementar API que fornece dados dos professores para o frontend • Implementar lógica de busca de professores pelo lado do cliente 	24/10/2023	31/10/2023	Melhora na aparência da tela de busca. Complemento de informações para o banco de dados. Logica de busca completa.	Todos	100%
Sprint 8	<ul style="list-style-type: none"> • Desenvolver cabeçalho com informações do professor. • Implementar funcionalidade de redirecionamento para a aplicação após autenticação de usuário bem sucedida. 	31/10/2023	07/11/2023	Desenvolvimento da tela de perfil dos professores, melhora na autenticação e redirecionamento.	Todos	100%
Sprint 9	<ul style="list-style-type: none"> • Criar pasta para registro de avaliações no banco de dados. • Realizar lógica de cálculo da nota do professor no Backend. • Implementar funcionalidade de avaliação 	07/11/2023	14/11/2023	Implementação de funcionalidades referentes a avaliação dos professores.	Todos	Em aberto

	dos professores.					
Sprint 10	<ul style="list-style-type: none"> Implementar página de matérias filtrada pela engenharia. Implementar página de professores que lecionam cada matéria. 	14/11/2023	21/11/2023	Implementação e melhorias em telas.	Todos	Em aberto
Sprint 11	<ul style="list-style-type: none"> Realizar teste unitários na aplicação. Deploy da aplicação. 	21/11/2023	28/11/2023	Teste da aplicação, correção e possíveis melhorias.	Todos	Em aberto
Sprint 12	<ul style="list-style-type: none"> Realizar testes unitários na aplicação. Entrega do produto. 	28/11/2023	05/11/2023	Teste da aplicação, correção e entrega.	Todos	Em aberto

2.4. Matriz de Comunicação

Tabela 4 – Comunicação do grupo

<i>Descrição</i>	<i>Área/Envolvidos</i>	<i>Periodicidade</i>	<i>Produtos Gerados</i>
<ul style="list-style-type: none"> Acompanhamento das Atividades em Andamento 	<ul style="list-style-type: none"> Equipe 	<ul style="list-style-type: none"> Semanal 	<ul style="list-style-type: none"> Ata de reunião Relatório de situação do projeto Novas Atividades Revisão
<ul style="list-style-type: none"> Acompanhamento dos Riscos, Compromissos, Ações Pendentes, Indicadores 	<ul style="list-style-type: none"> Equipe 	<ul style="list-style-type: none"> Semanal 	
- Comunicar situação do projeto	<ul style="list-style-type: none"> Equipe Monitor 	<ul style="list-style-type: none"> Semanal 	<ul style="list-style-type: none"> Ata de reunião, e Relatório de situação do projeto

2.5. Gerenciamento de Riscos

Comentários falsos ou tendenciosos de estudantes: Este risco pode ser crítico, pois pode afetar diretamente a qualidade da avaliação dos professores e prejudicar a credibilidade do sistema.

- **Mitigação:** Implementar ferramentas de detecção de comentários falsos ou tendenciosos e incentivar a moderação de conteúdo.
- **Contingência:** Implementar um sistema de denúncia e eliminação de comentários.

Falhas no design de segurança: A exposição de dados dos alunos é um risco significativo, já que a segurança dos dados é uma preocupação fundamental em sistemas educacionais.

- **Mitigação:** Realizar uma avaliação rigorosa da segurança do sistema, implementar práticas de criptografia e controle de acesso.
- **Contingência:** Ter planos de resposta a incidentes em vigor e notificar imediatamente as partes afetadas em caso de violação de segurança.

Não coletar feedbacks de usuários reais suficientes: A usabilidade é importante, mas este risco pode ser gerenciado de forma mais flexível, coletando feedback ao longo do tempo e ajustando o sistema conforme necessário.

- **Mitigação:** Implementar uma estratégia de coleta de feedback ativa e incentivadora para os usuários.
- **Contingência:** Realizar testes de usabilidade frequentes e ajustes iterativos com base no feedback disponível.

Integração Complexa: Problemas técnicos na integração podem atrasar o projeto, mas podem ser mitigados por meio de testes rigorosos e planejamento adequado.

- **Mitigação:** Realizar testes rigorosos de integração durante o desenvolvimento e envolver especialistas técnicos na resolução de problemas.
- **Contingência:** Ter um plano de contingência para lidar com atrasos na integração e manter as partes interessadas informadas sobre quaisquer impactos no cronograma.

Resistência pela parte dos professores: Embora seja um risco, a resistência pode ser gerenciada por meio de treinamento, comunicação eficaz e envolvimento dos educadores no processo de implementação.

- **Mitigação:** Envolver os professores no processo de planejamento e comunicar os benefícios da implementação do sistema.
- **Contingência:** Estabelecer um plano de comunicação eficaz para lidar com a resistência, oferecendo suporte contínuo.

2.6. Critérios de Replanejamento

- **Mudanças nos Requisitos:** Se houver mudanças nos requisitos do projeto que impactem o escopo, prazo ou recursos necessários, deverá ser replanejado para ajustar o projeto de acordo com as novas especificações.
- **Riscos do Projeto:** Os critérios de replanejamento estarão fortemente associados aos riscos identificados no projeto. Se um risco se materializar ou se tornar mais provável, o replanejamento será necessário para diminuir seus impactos e manter o projeto no caminho certo. De acordo com a definição de prioridades da equipe (Segurança > Usabilidade > Interface), em caso de risco à segurança da aplicação como todo, o replanejamento deverá ser feito imediatamente e a solução tratada como prioridade máxima; casos de usabilidade serão encarados de acordo com seu grau de importância para que o MVP seja alcançado e tratados, em primeiro momento, como débito técnico, o que pode evoluir para um replanejamento caso o tempo estimado para contenção do risco ultrapasse a duração total de 1 (uma) sprint (7 dias). Riscos de interface serão tratados como débito técnico e serão resolvidos sem necessidade de replanejamento.
- **Atrasos:** Se o projeto sofrer atrasos que possam comprometer o cronograma, será necessário replanejar para reavaliar e ajustar as datas de entrega.

3. Processo de desenvolvimento de Software

A equipe de desenvolvimento de software optou por adotar uma abordagem híbrida, estruturando seu processo por meio da combinação do framework Scrum, que oferece diretrizes sólidas para a gestão do projeto, com práticas técnicas do Extreme Programming (XP), que visam garantir a mais alta qualidade técnica do software. Essa escolha baseia-se nas melhores práticas definidas pelo Project Management Institute (PMI) no seu renomado Guia PMBOK® (2017), e busca otimizar o desenvolvimento do projeto, assegurando a eficiência na gestão e a excelência na execução técnica.

Dessa forma, o Scrum fornece a estrutura ideal para o planejamento, acompanhamento e entrega das metas do projeto, enquanto as práticas técnicas do XP, tais como pair programming e integração contínua, são adotadas para garantir que o software seja desenvolvido e testado com o mais alto padrão de qualidade técnica. O resultado é um processo que une a agilidade e a disciplina necessárias para atingir os objetivos do projeto, alinhados com as diretrizes do PMI.

Papéis:

- **Scrum Master:** Responsável por garantir que a equipe siga os princípios do Scrum, remove impedimentos e facilita a colaboração.
- **Product Owner:** Representa os interesses dos stakeholders, define as prioridades do backlog e toma decisões sobre o produto.
- **Equipe de Desenvolvimento:** Responsável por criar o produto de acordo com as prioridades estabelecidas.

Reuniões:

- **Periódicas:** A equipe decidiu optar por realizar minirreuniões durante o decorrer da sprint para acompanhar o progresso do grupo nas atividades e tratar possíveis obstáculos.
- **Sprint Review:** Ao final de cada sprint, é realizada uma reunião para formalizar as entregas e compreender possíveis atrasos. Após essa etapa, planejamos as atividades para a próxima sprint.

4. Detalhamento de atividades do projeto

4.1. Sprint 0

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Definição do produto.	SCRUM/XP	Teams	12/09

4.2. Sprint 1

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Definir tecnologias.	SCRUM /XP	Teams	19/09
Treinamento das equipes.	SCRUM /XP	Teams	19/09

4.3. Sprint 2

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Prototipação das telas	SCRUM /XP	Figma; Teams.	26/09

4.4. Sprint 3

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Codificar tela de login	SCRUM /XP	Visual Studio Code; React; Teams.	03/10
Codificar tela de cadastro	SCRUM /XP	Visual Studio Code; React; Teams.	03/10
Criar banco de dados	SCRUM /XP	Visual Studio Code; MongoDB; Teams.	03/10
Criar API	SCRUM /XP	Visual Studio Code; Node.js; Express; Teams.	03/10

4.5. Sprint 4

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
------------------	---------------	-------------------	----------------

Criar CRUD de usuário	SCRUM /XP	Visual Studio Code; MongoDB; Express; Node.js; Teams.	10/10
Alimentar DB de professores	SCRUM /XP	Visual Studio Code; MongoDB; Express; Node.js; Teams.	10/10
Implementar autenticação de usuário	SCRUM /XP	Visual Studio Code; React; MongoDB; Express; Node.js; Teams.	10/10
Iniciar documentação no MkDocs	SCRUM /XP	Visual Studio Code; Teams.	10/10

4.6. Sprint 5

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Implementar registro de usuário	SCRUM /XP	Visual Studio Code; React; MongoDB; Express; Node.js; Teams.	17/10
Integração Backend-Frontend	SCRUM /XP	Visual Studio Code; React; MongoDB; Express; Node.js; Teams.	17/10
Implementar tela inicial	SCRUM /XP	Visual Studio Code; React; Teams.	17/10

4.7. Sprint 6

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Otimizar conexão com o banco de dados	SCRUM /XP	Visual Studio Code; React; Teams.	24/10
Implementar API de professores	SCRUM /XP	Visual Studio Code; React; Teams.	24/10

4.8. Sprint 7

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Implementar API que fornece dados dos professores para o front end.	SCRUM /XP	Visual Studio Code; React; Teams.	31/10
Implementar logica de busca de professores pelo lado do cliente.	SCRUM /XP	Visual Studio Code; React; Teams.	31/10

4.9. Sprint 8

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Implementar funcionalidade de redirecionamento.	SCRUM /XP	Visual Studio Code; React; Teams.	07/11

4.10. Sprint 9

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Realizar lógica de cálculo da nota do professor no Backend.	SCRUM /XP	Visual Studio Code; React; Teams.	14/11
Implementar funcionalidade de avaliação dos professores.	SCRUM /XP	Visual Studio Code; React; Teams.	14/11

4.11. Sprint 10

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
------------------	---------------	-------------------	----------------

Implementar página de matérias filtrada pela engenharia.	SCRUM /XP	Visual Studio Code; React; Teams.	21/11
Implementar página de professores que lecionam cada matéria.	SCRUM /XP	Visual Studio Code; React; Teams.	21/11

4.12. Sprint 11

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Realizar teste unitários na aplicação.	SCRUM /XP	Visual Studio Code; jasmine; Teams.	28/11
Deploy da aplicação.	SCRUM /XP	Visual Studio Code; GitHub; Teams.	28/11

4.13. Sprint 12

<i>Atividade</i>	<i>Método</i>	<i>Ferramenta</i>	<i>Entrega</i>
Realizar testes unitários na aplicação.	SCRUM /XP	Visual Studio Code; jasmine; Teams.	05/11
Entrega do produto	SCRUM /XP	Visual Studio Code; GitHub; Teams.	05/11

5. LIÇÕES APRENDIDAS

5.1. Unidade 1

Acompanhamento Regular: Realize reuniões de acompanhamento diárias ou regulares para verificar o progresso das tarefas da sprint. Isso ajuda a identificar atrasos com antecedência.

5.2. Unidade 2

Comunicação Efetiva: Promova uma comunicação aberta e transparente dentro da equipe. Isso inclui relatar prontamente qualquer obstáculo ou problema que possa afetar o andamento do projeto.

5.3. Unidade 3

Durante o desenvolvimento da parte do back-end de nosso aplicativo, encontramos vários desafios, sendo o principal deles relacionado à implementação do banco de dados MongoDB e às operações CRUD relacionadas ao sistema de login

5.4. Unidade 4

Durante o desenvolvimento do projeto React, uma das principais dificuldades que enfrentamos foi a configuração das rotas entre as diferentes páginas do aplicativo utilizando a biblioteca React Router.

6. PRÓXIMOS PASSOS

7. REFERÊNCIAS BIBLIOGRÁFICAS

PMI - PROJECT MANAGEMENT INSTITUTE. Guia PMBOK®: Um Guia do Conhecimento em Gerenciamento de Projetos. 6. ed. Newtown Square, Pensilvânia: PMI, 2017.

Audy, Jorge. Scrum 360: Um guia completo e prático de agilidade. São Paulo: Casa do Código, 2015.