

# Grupo Fehu - Nome do Produto

## VISÃO DO PRODUTO E DO PROJETO

Versão 0.1

Tabela 1: Integrantes do Grupo

Matrícula	Nome	Função (responsabilidade)
222008459	DANIEL FERNANDES SILVA	
231012058	DANIELLE SOARES DA SILVA	
231026821	GABRIELA DOURADO FRANCA	
231011471	HENRIQUE DE OLIVEIRA BERNARDES	
231026429	JOSÉ AUGUSTO ALVES DE MORAES	Desenvolvedor
231012272	LETICIA ARISA KOBAYASHI HIGA	
222037577	MARIA EDUARDA QUARESMA DE ANDRADE	
231027005	MARIA SAMARA ALVES SILVA	
232029210	MARIANA PEREIRA DA SILVA	
231026545	PEDRO HENRIQUE FREIRE RODRIGUES	

Tabela 2: Histórico de Revisões

Data	Versão	Descrição	Autor(es)
14/11/2024	0.1	Redação inicial do capítulo 1, pendente alinhamento de ideias com a equipe.	José Augusto
15/11/2024	0.2	Instanciação do diagrama de ciclo de vida, justificativa da excolha do ScrumXP como metodologia e composição da organização do projeto	Henrique Bernardes e Letícia Arisa.
22/11/2024	0.3	Escrita inicial do capítulo 3, descrevendo o processo de desenvolvimento de software	José Augusto, Gabriela Dourado



## Sumário

<b>1</b>	<b><i>VISÃO GERAL DO PRODUTO.....</i></b>	<b><i>4</i></b>
1.1	Problema .....	4
1.2	Declaração de Posição do Produto .....	4
1.3	Objetivos do Produto .....	4
1.4	Tecnologias a Serem Utilizadas .....	4
<b>2</b>	<b><i>VISÃO GERAL DO PROJETO.....</i></b>	<b><i>6</i></b>
2.1	Ciclo de vida do projeto de desenvolvimento de software.....	6
2.2	Organização do Projeto .....	7
2.3	Planejamento das Fases e/ou Iterações do Projeto .....	7
2.4	Matriz de Comunicação .....	8
2.5	Gerenciamento de Riscos .....	8
2.6	Critérios de Replanejamento .....	8
<b>3</b>	<b><i>PROCESSO DE DESENVOLVIMENTO DE SOFTWARE .....</i></b>	<b><i>10</i></b>
<b>4</b>	<b><i>DECLARAÇÃO DE ESCOPO DO PROJETO .....</i></b>	<b><i>12</i></b>
4.1	Backlog do produto.....	12
4.2	Perfis.....	12
4.3	Cenários .....	12
4.4	Tabela de Backlog do produto .....	13
<b>5</b>	<b><i>REFERÊNCIAS BIBLIOGRÁFICAS .....</i></b>	<b><i>14</i></b>

# VISÃO DO PRODUTO E PROJETO

## 1 VISÃO GERAL DO PRODUTO

### 1.1 Problema

A empresa Catavento, cliente da equipe Fehu, é especializada na manufatura e comercialização de produtos para festas, com destaque para os bolos "fakes". No momento, o processo produtivo desses bolos é gerenciado de maneira informal e manual, utilizando uma planilha de demandas. Esta planilha é impressa e distribuída entre os departamentos responsáveis. Após a conclusão das demandas, os funcionários registram as informações diretamente na planilha, o que pode gerar inconsistências e dificultar o acompanhamento do processo.

Além disso, a cliente mencionou situações em que os funcionários, conscientes de todas as demandas futuras, tendem a priorizar a entrega de tarefas mais simples em detrimento das que exigem maior esforço.

Nesse contexto, surge a oportunidade de implementar uma solução tecnológica que centralize o controle das demandas. Essa abordagem permitirá que:

- Gerentes de cada departamento tenham acesso em tempo real ao status de cada demanda, assegurando um melhor controle da produção.
- Funcionários possam visualizar apenas a demanda atual, evitando a priorização de tarefas específicas em detrimento de outras, o que promoverá uma gestão mais equilibrada e eficiente das atividades.

### 1.2 Declaração de Posição do Produto

Após a identificação do problema e da oportunidade de solução mencionados anteriormente, apresentamos no Quadro 1, a seguir, a declaração de posição do produto. Esta declaração compila as informações essenciais sobre a visão da equipe em relação ao produto a ser desenvolvido, as necessidades que ele atende e sua posição no mercado atual.

**Quadro 1: Declaração de Posição do Produto**

<b>Para:</b>	Empresa Catavento
<b>Necessidade:</b>	uma solução tecnológica para a gerência de demandas
<b>O (nome do produto):</b>	[nome do produto] é um App mobile
<b>Que:</b>	centraliza as demandas de cada departamento, permitindo o lançamento e acompanhamento de demandas em tempo real pelos gerentes e a apresentação das demandas em forma estritamente sequencial aos funcionários
<b>Ao contrário:</b>	tecnologias voltadas para o gerenciamento de projetos como o Trello ou o Slack poderiam substituir o método atual de gerenciamento
<b>Nosso produto:</b>	porém, o [nome do produto] será uma ferramenta especializada em atender nas necessidades da cliente, adaptada ao contexto produtivo da empresa.

### 1.3 Objetivos do Produto

O aplicativo tem como principal objetivo otimizar o gerenciamento das demandas do processo produtivo dos bolos "fakes". Ele substituirá a abordagem atual, que utiliza uma planilha impressa, por uma solução tecnológica que facilitará o controle da produção.

Adicionalmente, o aplicativo será capaz de gerar relatórios de desempenho para cada funcionário, o que permitirá um cálculo mais eficiente dos bônus por atingimento de metas específicas e contribuirá para uma gestão mais eficaz da equipe.

### 1.4 Tecnologias a Serem Utilizadas

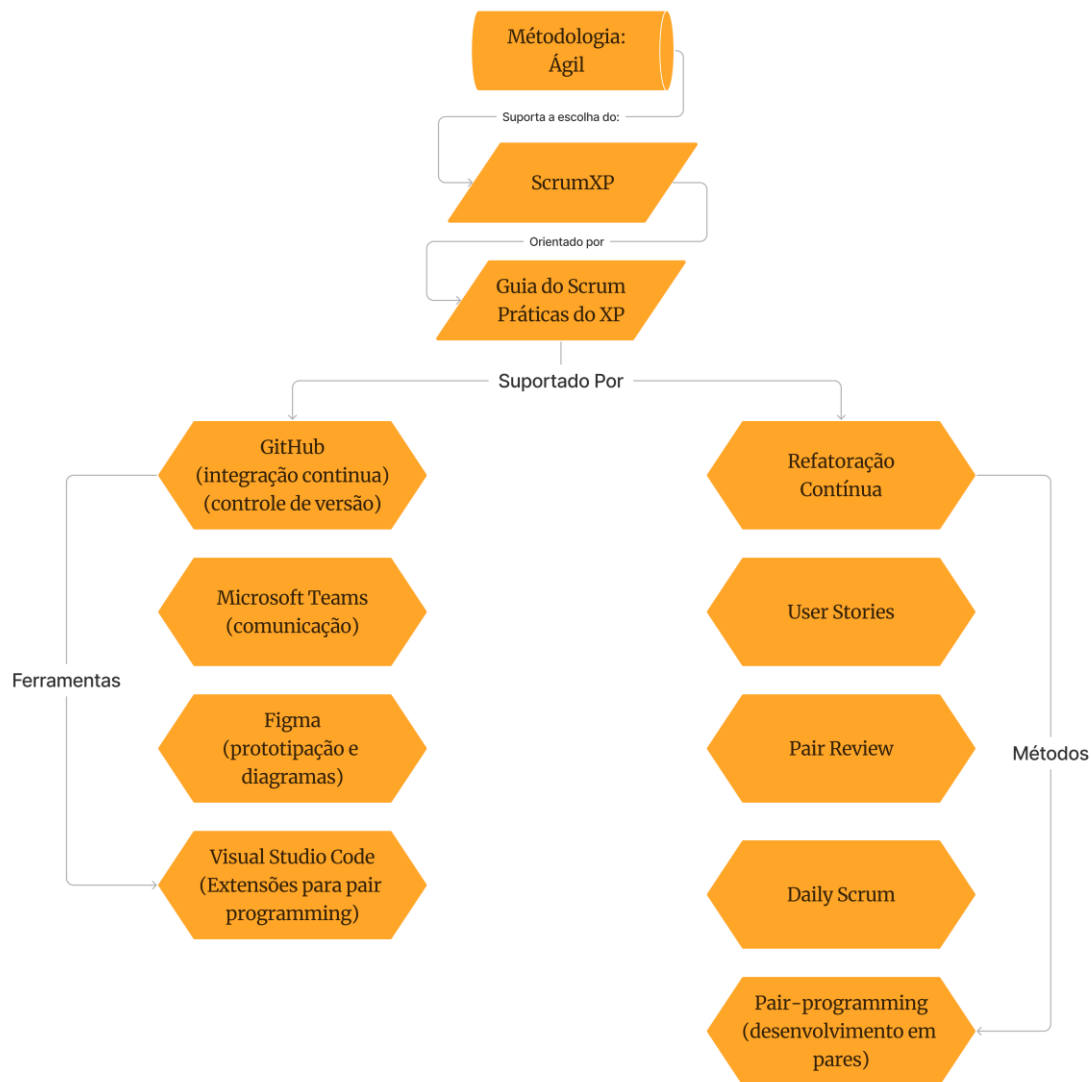
Para o desenvolvimento do projeto, serão empregadas as seguintes tecnologias:

- Linguagem Dart: Esta é a principal linguagem utilizada, tanto para a definição do frontend através do Flutter, quanto para a implementação da lógica de negócios e a comunicação com o Supabase.
- Flutter: Um framework de desenvolvimento mobile multiplataforma, que será utilizado na construção do frontend da aplicação.
- Supabase: Um serviço de backend que oferece diversas ferramentas, incluindo comunicação com um banco de dados PostgreSQL e autenticação de usuários com diferentes níveis de acesso, entre outros recursos.

## 2 VISÃO GERAL DO PROJETO

### 2.1 Ciclo de vida do projeto de desenvolvimento de software

Figura 1: Ciclo de Vida do Projeto



Fonte: autoria própria (2024)

Primeiramente, o desenvolvimento em prazo limitado (1 semestre) exige a escolha de uma metodologia ágil que permita entregas rápidas e flexibilidade para ajustes ao longo do tempo, além de acomodar mudanças de forma aceitável. De acordo com Wonohardjo et al. (2019):

Scrum is often used to face complex problems in software development process and has been proven to increase productivity and reducing software development cost. Scrum deliver the software in iterations, that means the software is delivered in partial, so any changes or problems can be handled before all parts of software is ready to deliver. (Wonohardjo et al., 2019)

Dessa forma, o produto deve ser entregue de maneira incremental, trabalhando e lançando versões ao longo do desenvolvimento para receber feedback e realizar ajustes necessários antes da versão final.

A solução proposta é um aplicativo mobile desenvolvido com Flutter, sendo possível compilar o aplicativo para Android e iOS a partir de uma base de código única, agilizando o desenvolvimento. O backend será suportado por Dart e o banco de dados pelo Supabase. A

escolha da linguagem Dart se alinha ao uso do Flutter, facilitando a criação de uma interface com compatibilidade para as funcionalidades do frontend.

A metodologia ScrumXP foi escolhida por combinar a estrutura do Scrum, que permite ciclos curtos e feedback frequente, com as práticas do Extreme Programming (XP), como o "Pair Programming" e a integração contínua. De acordo com Sahota (2011), conforme citado por Dada (2022), as práticas Scrum precisam ser integradas ao XP para garantir um projeto bem-sucedido, embora existam práticas que se sobrepõem entre os dois métodos. Assim, com sprints definidas de 10 dias, é possível lançar incrementos rapidamente, receber feedback e ajustar o desenvolvimento conforme necessário. As práticas do XP, como a refatoração contínua e a programação em pares ajudam no desenvolvimento e a manter o código organizado. A flexibilidade proporcionada pelo ScrumXP permite adaptar o projeto a novos requisitos ou alterações de prioridades, maximizando o valor entregue em um semestre para a cliente.

O projeto será organizado por meio de histórias de usuário, que capturam as necessidades dos usuários e as traduzem em funcionalidades. Reuniões diárias (Daily Scrum) ajudam a manter o progresso alinhado e identificar dificuldades, e ao final de cada sprint serão realizadas Sprint Reviews para revisar o progresso e retrospectivas para identificar o que pode ser melhorado.

As ferramentas escolhidas para o gerenciamento e desenvolvimento incluem o Figma, para organizar o backlog, acompanhar o progresso das tarefas, composição de diagramas e protótipos, e o GitHub, para controle de versão e integração contínua. O Microsoft Teams será utilizado para comunicação rápida entre a equipe, Visual Studio Code para programação em Dart e Flutter e programação em pares com extensões.

Logo, conclui-se que a escolha do ScrumXP é apropriada devido à necessidade de ciclos curtos e feedback contínuo, essenciais para o projeto devido à curta duração e com requisitos mutáveis, além dos métodos e ferramentas suportarem um desenvolvimento incremental, ágil e prático.

## 2.2 Organização do Projeto

<i>Papel</i>	<i>Atribuições</i>	<i>Responsável</i>	<i>Participantes</i>
Desenvolvedor	Codificar o produto, codificar testes unitários, realizar refatoração	Daniel, Danielle, Gabriela, Henrique, José, Leticia, Maria Eduarda, Maria Samara, Mariana, Pedro	Daniel, Danielle, Gabriela, Henrique, José, Leticia, Maria Eduarda, Maria Samara, Mariana, Pedro
Dono do Produto	Atualizar o escopo do produto, organizar o escopo das sprints, validar as entregas	Henrique	Henrique
Analista de Qualidade	Garantir a qualidade do produto, garantir o cumprimento do conceito de pronto, realizar inspeções de código, executar testes, identificação de erros e defeitos, planejamento de testes	Maria Eduarda, Mariana	Maria Eduarda, Mariana
Analista de Requisitos	Fazer o levantamento dos requisitos, analisar e documentar os requisitos, priorizar os requisitos de acordo com o objetivo do produto, gerenciar as mudanças de requisitos	Henrique, Maria Eduarda, Pedro	Henrique, Maria Eduarda, Pedro
Designer de Protótipos	Criar modelos iniciais do produto, fazer a validação das do design, interações e funcionalidades	Danielle, Leticia, Mariana, Maria Samara	Danielle, Leticia, Mariana, Maria Samara
Cliente	Validar os requisitos e protótipos, fornecer feedbacks	Poliana	Poliana

## 2.3 Planejamento das Fases e/ou Iterações do Projeto

*[Registrar o projeto, as fases de seu ciclo de vida e suas iterações, especificando suas datas de início e de fim, bem como os produtos a serem gerados.]*

*O planejamento do projeto deve ser atualizado de maneira sucessiva, a partir da realização de cada ciclo (sprint, iteração) de desenvolvimento]*

<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>
---------------	--------------------------	--------------------	-----------------	------------------------	---------------------	--------------------

<i>Sprint 1</i>	<i>Definição do Produto</i>	<i>dd/mm/aaaa</i>	<i>dd/mm/aaaa</i>	<i>(*)</i>	<i>Os papeis envolvidos é uma boa estratégia</i>	<i>Intervalos mínimos de 10 em 10 %</i>
<i>Sprint 2</i>	<i>MVP e Planejamento do Projeto</i>	<i>dd/mm/aaaa</i>	<i>dd/mm/aaaa</i>			
<i>Sprint 3</i>	<i>Funcionalidades A, B, C, D</i>	<i>dd/mm/aaaa</i>	<i>dd/mm/aaaa</i>			
<i>Sprint 4</i>	<i>Funcionalidades E, F e G</i>	<i>dd/mm/aaaa</i>	<i>dd/mm/aaaa</i>			
<i>...</i>	<i>...</i>	<i>...</i>	<i>...</i>			

(\*): Podem ser apelidos (Aliases), desde que existam suas definições completas, após o quadro.

## 2.4 Matriz de Comunicação

[Esta seção descreve a estratégia de comunicação adotada para monitoramento do progresso do projeto. Identificar a periodicidade de reuniões e o envio dos relatórios exigidos pelo processo e opcionalmente outros relatórios exigidos pelo cliente.]

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

## 2.5 Gerenciamento de Riscos

[O Gerenciamento de Riscos consiste em tarefas, como:

- Identificar todos os riscos possíveis e detectáveis em cada fase do projeto;*
- Executar as ações para mitigar os riscos que tenham um alto grau de exposição ao risco caso este ocorra na Lista de Riscos do Projeto;*
- Fazer uma revisão da lista dos riscos periodicamente, com o propósito de averiguar uma possível incidência de um risco e ver se há outros riscos ainda não relatados;*
- Em caso de confirmação de um risco previsto, agir no sentido de contingenciá-lo conforme programado;*
- Registrar os riscos no Painel de Controle do Projeto e no Plano do Projeto (Riscos iniciais);]*

***Os riscos do projeto devem ser acompanhados e atualizados periodicamente.***

***Sugestão:*** construir um quadro contendo: o risco, grau de exposição, mitigação (se necessária, pois depende do grau de exposição) e plano de contingência.<sup>1</sup>

## 2.6 Critérios de Replanejamento

[Descrever os critérios de replanejamento que serão utilizados, caso seja necessário realizá-lo no projeto.]

*Observações*

- Fontes de necessidades de replanejamento importantes são os riscos do projeto. por isso é tão importante manter sua gestão de riscos em dia. Os critérios de replanejamento apoiados por riscos, devem identificar quais deles podem causar esse efeito e devem estar alinhados aos planos de mitigação e/ou contingência***

<sup>1</sup> Lembrando: plano de mitigação existe para tentar evitar que um risco ocorra. Plano de contingência é acionado se um risco efetivamente ocorre.



- *Os critérios de replanejamento do projeto devem ser acompanhados e atualizados a cada ciclo. E, aplicados, conforme necessidade. Replanejamentos do projeto ou qualquer outra alteração causa o versionamento deste documento.*

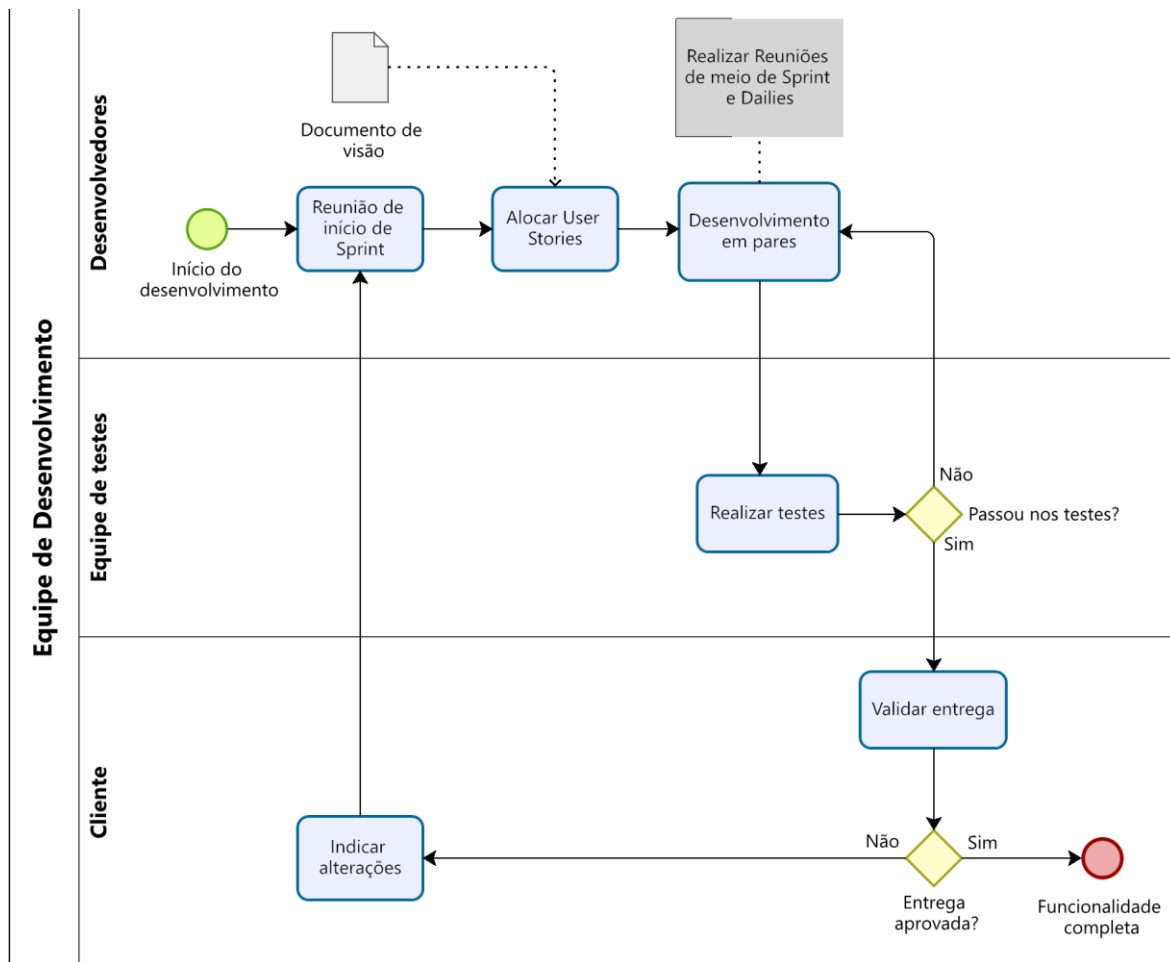
### 3 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

Algumas ferramentas serão utilizadas no decorrer do processo de desenvolvimento de software, assim será preciso o uso de algumas ferramentas como foram apresentadas na seção 2.1. A metodologia utilizada será o Scrum, com o intuito de feedbacks frequentes e ciclos curtos e práticas do Extreme Programming (XP), assim o ScrumXP.

Assim, com a utilização do ScrumXP e as ferramentas escolhidas temos uma certa divisão em cada parte do processo de desenvolvimento, explicada em detalhes abaixo e ilustrada na Figura 2 a seguir.

- Reunião de sprints e dailies: O Teams será o meio de comunicação para reuniões para tomada de decisões, alinhamentos de ideias e alterações. Além da ferramenta *WhatsApp* para comunicações mais rápidas.
- Desenvolvimento: O desenvolvimento terá o uso de várias ferramentas, entre elas:
- Backend – Linguagem Dart e o banco de dados pelo Supabase;
- Frontend – Figma, Flutter e outras ferramentas de estilização.
- Realizar testes: Inicialmente, serão realizados testes de unidade no backend para validar funcionalidades específicas, assegurando que cada parte do sistema funcione de forma independente. Em seguida, os testes de integração verificarão a comunicação entre o backend (incluindo o Supabase) e o frontend, garantindo que os dados sejam transmitidos e manipulados corretamente. O frontend será testado manualmente para avaliar o comportamento das funcionalidades, navegação e experiência do usuário. Ferramentas como o Figma serão usadas para confirmar que o design está de acordo com o aprovado, garantindo uma interface coesa e usável. Ao final de cada sprint, testes de aceitação serão realizados junto à cliente para validar os requisitos implementados e alinhar possíveis ajustes antes de avançar para a próxima etapa.
- Validar entrega: Teremos uma reunião com a cliente para que possamos alinhar as ideias com a entrega de cada requisito e com a sua aprovação seguir em frente com o desenvolvimento, caso haja uma reprovação, anotamos as sugestões da cliente e nos reuniremos para poder alinhar as nossas novas ideias com as da cliente.

**Figura 2: Fluxo de Desenvolvimento de Software**



Fonte: autoria própria (2024)

## 4 DECLARAÇÃO DE ESCOPO DO PROJETO

### 4.1 Backlog do produto

*Forneça o backlog do produto:*

- *Sugiro as tabelas, dos itens a seguir organizadas em perfis, cenários, contendo cada um deles, os requisitos obrigatórios (MUST), e desejáveis (Should ou Could). As tabelas que se seguem são sugestões mínimas. Assim sendo elas podem ser modificadas/adaptadas pela equipe de acordo com as necessidades de cada desenvolvimento, desde que justificadas.*
- *Defina o que cada uma dessas classes representa para o software em desenvolvimento que justifique sua classificação (visão da equipe a ser validada pelo P.O.)*
- *Identifique a forma de obtenção do requisito (elicitação): brainstorm, entrevistas, questionários, etc.*

### 4.2 Perfis

*Perfis de acesso ao sistema de software podem ser importantes para identificar funcionalidades disponíveis de acordo com tais perfis. Neste sentido eles precisam ser identificados para, mais tarde, serem tratados as limitações de cada um deles. Além disso, é importante, ao cadastrar usuários que tenham permissões de acesso diferenciadas, que elas também sejam identificadas. Para isso é sugerida a seguinte tabela.*

**Tabela : Perfis de acesso**

#	Nome do perfil	Características do perfil	Permissões de acesso
<I>	<ex> Administrador	Reponsável por manter os perfis de acesso da aplicação, criar novos usuários, alterar usuários já existentes, ou excluir usuários (Manter usuários)	Descrever sucintamente as permissões de acesso

*Perfis diferentes podem usar técnicas diferentes para a elicitação dos requisitos. Descreva e justifique como parte deste item, qual(is) técnicas de elicitação serão usadas (criem tabelas para isso).*

### 4.3 Cenários

**Tabela : Cenários funcionais**

SistemaGa: xxx – Cenários funcionais		
Numeração do cenário	Nome do cenário	Sprints

*Alternativamente/ ou adicionalmente pode ser usado um mapa hierárquico para representar os cenários e suas Sprints e itens do backlog relacionados.*

#### 4.4 Tabela de Backlog do produto

Tabela : Backlog do produto

[illegible]

## 5 REFERÊNCIAS BIBLIOGRÁFICAS

- WONOHARDJO, Eduard Pangestu; SUNARYO, Rizky Febriyanto; SUDIYONO, Yusuf. A systematic review of SCRUM in software development. JOIV: International Journal on Informatics Visualization, v. 3, n. 2, p. 108-112, 2019. Acesso em: 15 nov. 2024. Disponível em: <https://joiv.org/index.php/joiv/article/view/167>.
- DADA, Oluwaseun Alexander; SANUSI, Ismaila Temitayo. The adoption of Software Engineering practices in a Scrum environment. African Journal of Science, Technology, Innovation and Development, v. 14, n. 6, p. 1429-1446, 2022. Acesso em: 15 nov. 2024. Disponível em: <https://journals.co.za/doi/full/10.1080/20421338.2021.1955431>.
-