



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PÓS GRADUAÇÃO LATO SENSU RESIDÊNCIA EM DESENVOLVIMENTO DE SOFTWARE

ALUNOS: FLÁVIO MEDEIROS
JAMESON RIBEIRO
JOSÉ MATHEUS
RAUL CAVALCANTI
RODRIGO FERREIRA

TRABALHO REALIZADO PARA A
DISCIPLINA DE **FUNDAMENTOS**
DO PROCESSO ÁGIL DE
DESENVOLVIMENTO DE
SOFTWARE.

SUMÁRIO

1 - INTRODUÇÃO

- 1.1 - Apresentação do BMC
- 1.2 - Apresentação do PMC
- 1.3 - Apresentação do BackLog

2 - METODOLOGIAS

- 2.1 - Filosofia LEAN
- 2.2 - Gestão do Projeto - Scrum
- 2.3 - Planning Poker
- 2.4 - Gestão do desenvolvimento do produto
- 2.5 - Entregas
- 2.6 - Gestão da Qualidade
- 2.7 - Gestão dos riscos

3 - PLANEJAMENTO DA EXECUÇÃO DO PROJETO

- 3.1 - Sprint 01
- 3.2 - Sprint 02
- 3.3 - Sprint 03
- 3.4 - Sprint 04
- 3.5 - Sprint 05
- 3.6 - Lançamento do MVP

1 - INTRODUÇÃO

Este documento tem por objetivo apresentar o planejamento, estimativas, metodologias e técnicas usadas no decorrer do projeto para desenvolvimento de uma aplicação de To Do list.

A partir de observações, foi verificada uma janela de oportunidade no mercado de ferramentas de gestão de tarefas a fazer. Desta feita modelamos nossa ferramenta trazendo como principal diferencial das concorrentes, sua interação por meio de uma assistente virtual. O canvas do negócio pode ser visto abaixo:

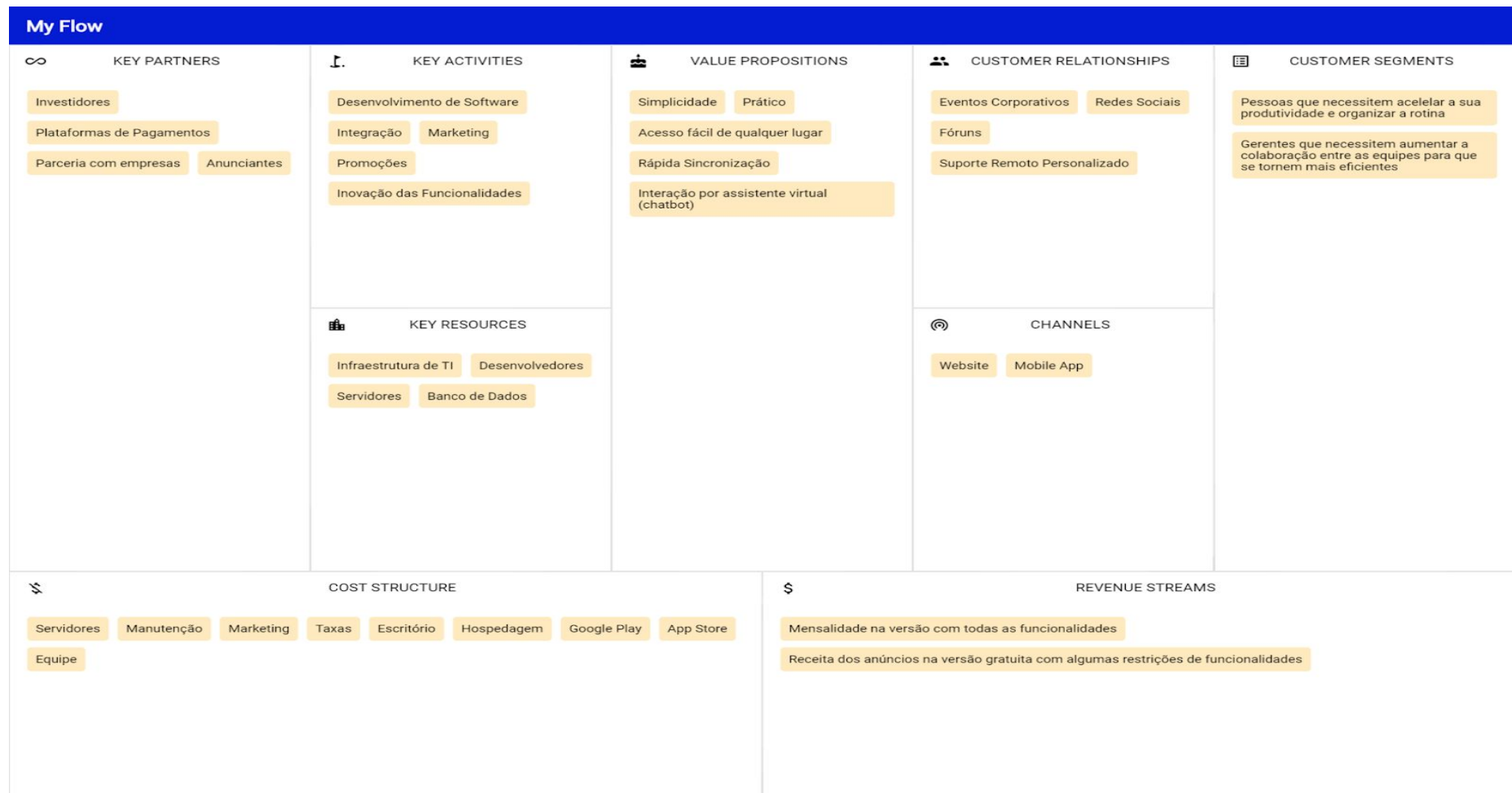


Figura 01 - Business Model Canvas

Com a definição do modelo de negócios, foi construído o canvas do projeto (figura 02) para através dessa ferramenta, elucidar e solidificar melhor pontos cruciais sobre a viabilidade do mesmo. O produto terá uma projeção tanto nas lojas de aplicativos, Android e IOS, bem como WEB, porém reduzimos o escopo do MVP à apenas a versão WEB, reduzindo tempo de desenvolvimento e custos, proporcionando um lançamento em curto prazo, para entendermos mais rapidamente a aceitação do produto.

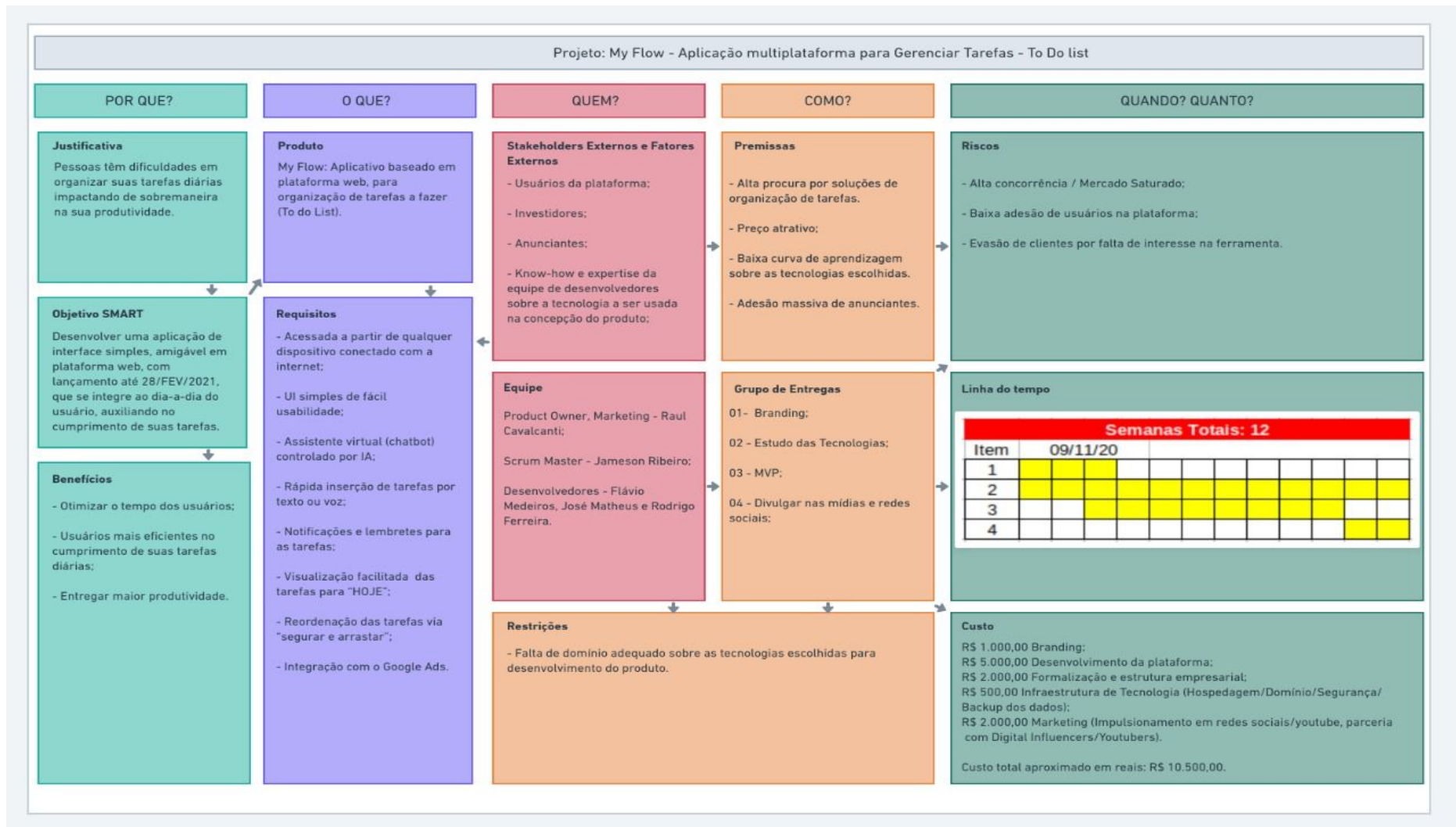


Figura 02 - Project Model Canvas

2 - METODOLOGIAS

O planejamento detalhado do desenvolvimento do MVP é explicado a seguir, com todas suas atividades e métodos que permeiam o processo.

2.1 - Filosofia LEAN

Adotamos como cultura empresarial o LEAN, filosofia de gestão enxuta como forma de otimizar os processos gerenciais de uma organização buscando a eliminação de desperdícios e melhoria contínua. Eis os 7 princípios de mentalidade corporativa que devem ser seguidos:

PRINCÍPIOS LEAN		
01	Eliminar o desperdício	Elimine qualquer forma de desperdício, tudo que não agrega valor deve ser eliminado: funcionalidades/códigos que não serão utilizadas, defeito, aquilo que não agrega valor, especificar todo o software antes de iniciar a construção.
02	Fortalecer o time	Criar um ambiente onde a equipe trabalhe de forma auto-organizada e auto-dirigida, evitando micro-gerenciamento.
03	Entregas rápidas	Reduza o tempo entre uma entrega e outra dos seus times adotando práticas ágeis. Isto permitirá aos times receber um feedback mais rápido do produto e aprender com os erros.
04	Otimizar o todo	Entender que o software concluído é muito mais que a soma das partes entregues e verificar como ele está alinhado com os objetivos da empresa.
05	Construir qualidade	Garantir qualidade no desenvolvimento do software utilizando técnicas como: Teste unitário, Refatoração:e Integração contínua.

06	Adiar decisões	As decisões devem ser tomadas no último momento possível, quando tivermos o maior volume de fatos e informações possíveis visando serem desta forma mais assertivas.
07	Amplificar conhecimento	Priorizar a comunicação e o feedback contínuos entre equipes e cliente final durante o processo de desenvolvimento de software.

Como forma de aplicar esta filosofia dentro da gestão do projeto, escolhemos o framework Scrum, por ter uma relação íntima com os princípios do LEAN, implementando então o modelo ágil como metodologia.

2.2 - Gestão do Projeto - Scrum

Essa metodologia ágil de gestão de projeto será utilizada seguindo de forma integral suas cerimônias, atividades e papéis. As Sprints terão como timebox padrão 15 dias (02 semanas), de segunda a sábado, começando na segunda-feira, pela manhã, da primeira semana com Sprint Planning , seguindo com atividades diárias de desenvolvimento do produto e reuniões de acompanhamento, finalizando no sábado da segunda semana com Sprint Review pela manhã e à tarde com Sprint Retrospective. Abaixo segue uma representação visual da Sprint:

Atividades da Sprint														
Legenda	Sprint Planning							Daily meeting						
	Sprint Retrospective							Sprint Review						
	Codificação (guiada conforme FDD - Feature Driven Development)													
Semana	01							02						
Dia	01 seg	02 ter	03 qua	04 qui	05 sex	06 sab	07 dom	08 seg	09 ter	10 qua	11 qui	12 sex	13 sab	14 dom

Manhã														
Tarde														

Ademais, ferramentas visuais serão utilizadas como forma de acompanhamento do andamento do projeto. O quadro Kanban (figura 03) vai focar no controle de status das users stories no deadline da Sprint, enquanto o gráfico Burndown (figura 04) apresentará uma visão do andamento geral do projeto.

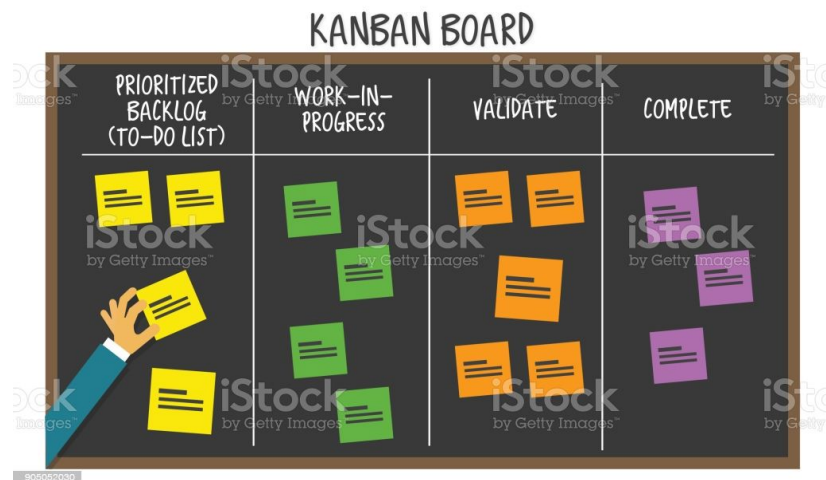


Figura 03 - Kanban Board

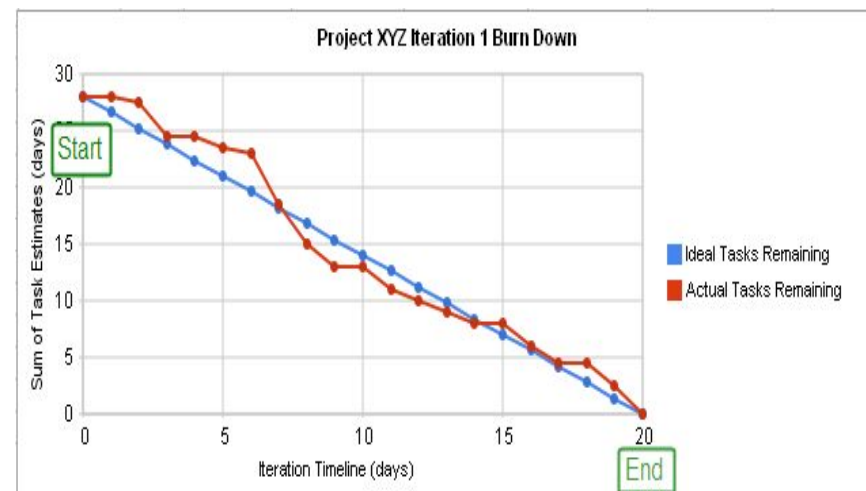


Figura 04 - Burndown Chart

Cada membro do time exercerá um papel preconizado no Scrum, bem como terá responsabilidades específicas quanto ao negócio. Abaixo segue um quadro apresentando esta divisão de funções:

Membro	Papel SCRUM	Função de negócio
Raul	Product Owner	Busca de investidores, fechamento de parcerias.
Jameson	Scrum Master	Gerente de Qualidade e Riscos
José Matheus	Desenvolvedor	Marketing, Designer
Flavio	Desenvolvedor	Construção do Backend e suas integrações.
Rodrigo	Desenvolvedor	Construção do Frontend e suas integrações.

2.3 - Planning Poker

Através dessa atividade buscamos priorizar as tarefas e fazer estimativas do esforço que é exigido para executá-las.

	ITEM	DESCRIÇÃO		JAMESON	FLAVIO	RODRIGO	MATHEUS
ITEM REFERENCIA (STORY POINTS = 0)	Como usuário, gostaria de poder acessar o sistema em qualquer dispositivo;	O sistema deverá ter versões nas plataformas web.		0	0	0	0
	Como usuário, gostaria de poder ter minhas rotinas/notas sincronizadas em tempo real.	O sistema deverá permitir que os usuarios possam se cadastrar, permitindo assim seus dados sejam sincronizados em todos os dispositivos que o mesmo mantiver login.		5	40	5	5
	Como usuário, gostaria que o sistema fosse de fácil compreensão e usabilidade.	O sistema deve ter uma interface intuitiva e acessível + funcionalidade reordenação de tarefas via segurar e arrastar.		13	40	8	8
	Como usuário, gostaria que o sistema me notificasse sobre meus prazos e tarefas	O sistema deve notificar o usuario, seja por email, push notification e/ou notificação sonora.		8	9	5	13
	Como usuário, gostaria que o sistema mostrasse um resumo das tarefas diarias.	O sistema deve ter proporcionar uma visualização facilitada das tarefas pra "hoje"		20	9	2	5
	Como usuário, gostaria de criar, editar e remover itens/tarefas no app.	o app deverá permitir que usuários possam criar e manusear tarefas por meio de assistente virtual (chatbot)		40	40	3	40

Figura 05 - resultado do Planning Poker

Resultado do backlog estimado em Story Points:

Para os itens com alta divergência de pontos apresentados pelos integrantes, uma média foi feita e o valor final da estimativa registrada.

id	Item Backlog	Descrição/Detalhes	Complexidade (Story Points)
1	Como usuário, gostaria de poder acessar o sistema em qualquer dispositivo;	<p>O sistema deverá ter versões nas plataformas web, mobile.</p> <p>Foi programado em acordo da equipe com o OP que para a primeira entrega será priorizado a versão web do aplicativo.</p> <p>IMPLEMENTAÇÃO: Versão WEB APP</p>	Por ser uma premissa, foi dado o valor 0 (zero), para que fosse usado como base para estimar as outras estórias.
2	Como usuário, gostaria de poder ter minhas rotinas/notas sincronizadas em tempo real.	<p>O sistema deverá permitir que os usuários possam se cadastrar, permitindo assim seus dados sejam sincronizados em todos os dispositivos que o mesmo mantiver login.</p> <p>Deverá ser implementado no sistema a funcionalidade de cadastro, + banco de dados para persistir os dados dos usuários e 'tasks to-do'.</p> <p>IMPLEMENTAÇÃO: Banco de dados, Cadastro de usuário,</p>	5
3	Como usuário, gostaria que o sistema fosse de fácil compreensão e usabilidade.	<p>O sistema deve ter uma interface intuitiva e acessível.</p> <p>Deve ser adotado em seu design, padrões minimalistas, com fácil manipulação das listas</p> <p>IMPLEMENTAÇÃO: Padrões de acessibilidade, funcionalidade reordenação de tarefas via segurar e arrastar.</p>	20
4	Como usuário, gostaria que o sistema me notificasse sobre meus prazos e tarefas	<p>O sistema deve notificar o usuário.</p> <p>seja por email, push notification e/ou notificação sonora.</p> <p>IMPLEMENTAÇÃO: Integração de notificação por email, Funcionalidade push notification em WEB App.</p>	9
5	Como usuário, gostaria que o sistema mostrasse um resumo das tarefas diárias.	<p>O sistema deve ter proporcionar uma visualização facilitada das tarefas pra "hoje"</p> <p>Deve apresentar um resumo do que estava programado para o dia.</p> <p>IMPLEMENTAÇÃO: Botão ou area reservada para o resumo das tarefas.</p>	9
6	Como usuário, gostaria de criar, editar e remover itens/tarefas no app.	<p>o app deverá permitir que usuários possam criar e manusear tarefas.</p> <p>Poderá ter interação por meio de assistente virtual (chatbot) o app deverá ter seu fluxo de funcionamento por meio de assistente virtual (chatbot).</p> <p>IMPLEMENTAÇÃO: Implementar logica do CRUD das tarefas, Integração de api para interação com chatbot.</p>	40

Figura 06 - Estimativa

2.4 - Gestão do Desenvolvimento de Software - FDD

Enquanto o Scrum tem como foco principal a organização de atividades visando a gestão do projeto, a metodologia FDD - Feature Driven Development, cuida do espectro de gestão do desenvolvimento do software.

O timebox de atuação do FDD será cada Sprint, ou seja, cada incremento no produto. Como entrada, o processo recebe uma lista de funcionalidades/tarefas a serem desenvolvidas, o Backlog da Sprint. Nesta etapa os envolvidos no processo resumem-se apenas a equipe de desenvolvedores, testadores, etc., que vão atuar diretamente na codificação. O FDD preconiza que o desenvolvedor é o único responsável pelo módulo que este desenvolve. Incentiva também o uso de testes do início ao fim do processo, além da utilização de integração contínua para diminuir riscos de incompatibilidade entre funcionalidades desenvolvidas por diferentes desenvolvedores. Como resultado do uso do FDD em nosso processo de desenvolvimento seguiremos as seguintes etapas:

- 1 - Desenvolvedor recebe uma das tarefas do backlog e fica responsável pela sua codificação;
- 2 - Seguindo o método TDD, codifica a partir de testes unitários e/ou de integração;
 - 2.1 - Um teste é criado para a funcionalidade de modo que inicialmente seja falho;
 - 2.2 - A funcionalidade é então desenvolvida até que passe no teste criado;
 - 2.3 - O código é refatorado para melhorar estruturas, diminuir as linhas e aprimorar a performance.
- 3 - Através da ferramenta Jenkins de integração contínua e testes automatizados, a nova funcionalidade é logo “mergeada” ao código principal.

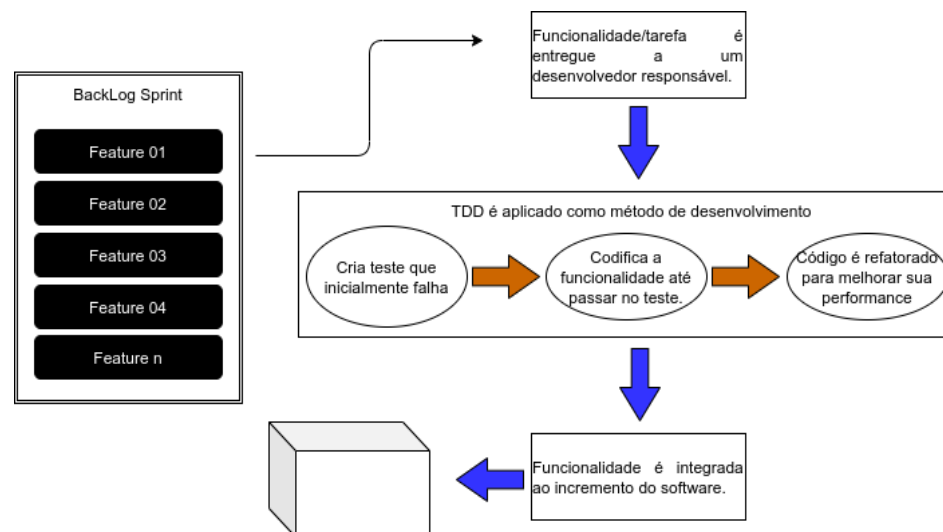


Figura 07 - Workflow

O controle visual do andamento da codificação de cada item do Backlog Sprint ficará a cargo da ferramenta Kanban, apresentando os seguintes status: A fazer, Fazendo, Testado, Concluído.

Em um cenário ideal, todas as funcionalidades previstas para aquela Sprint devem estar Concluídas ao final deste timebox.

2.5 - Entregas

Em um primeiro momento, será disponibilizado para o cliente final apenas a versão WEB da aplicação. A redução do escopo se dá com a finalidade de reduzir custos e tempo para o lançamento do produto no mercado, obtendo mais rapidamente o feedback de aceitação. A entrega se dará em 2 momentos distintos:

Ao final da Sprint 04, já teremos um release funcional do produto, que será disponibilizado para um grupo heterogêneo de aproximadamente 100 usuários com diferentes perfis de ocupação e gênero. Essa heterogeneidade se faz interessante para termos um panorama mais preciso nos feedbacks. Esse público alvo será selecionado dentre o círculo de amigos dos membros do time, investidores e parceiros. São eles:

- 50% de homens e 50% de mulheres;
- 20% de estudantes universitários;
- 20% de profissionais autônomos;
- 20% de trabalhadores de nível operacional;
- 20% de trabalhadores de nível gerencial (coordenadores, gerentes, diretores);
- 20 % de empresários.

Através de um questionário(Google Forms) <https://forms.gle/96CzY2wAAd9jYvT38> aplicado neste grupo, colheremos suas impressões sobre o produto e oportunidades de melhorias. O P.O receberá o feedback, procederá com uma análise sobre os dados coletados, levantará novas histórias de usuários, fará a priorização e junto ao time, incluirá as novas tarefas para desenvolvimento no backlog da Sprint 05.

Esquema da primeira entrega		
Dia	0	1 ao 7

Atividade	Disponibilização da versão de avaliação do produto em plataforma gratuita (HEROKU). Comunicação ao público alvo e liberação de acesso.	Uso da aplicação pelos clientes
		Coleta de feedbacks através do formulário
		Levantamento e priorização de novas tarefas.
		Inclusão no backlog da Sprint 05.

A segunda entrega será o lançamento oficial do MVP, ao final da Sprint 05, com todas as funcionalidades implementadas.

2.6 - Gestão da Qualidade

Como forma de garantir a solução dos possíveis problemas que poderiam surgir durante toda a fase do projeto e permitir uma melhoria contínua, adotamos o método PDCA como ferramenta de gestão da qualidade. O método PDCA permite uma se tenha uma boa organização e clareza para o entendimento e aplicação de melhorias no processo. Sua aplicação consiste em 4 fases:

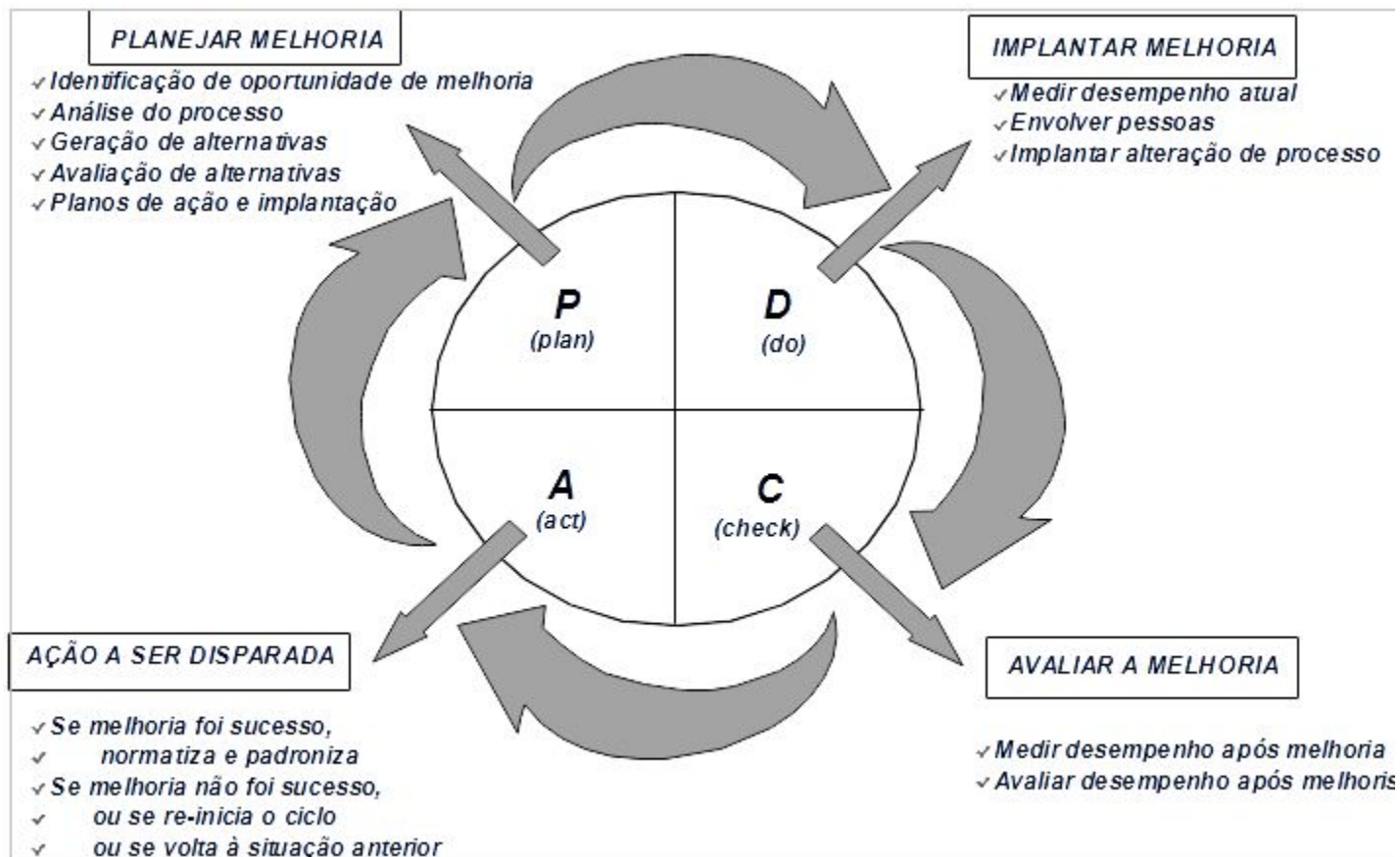
Plan: Nessa primeira fase identificamos o problema a ser trabalhado ou resultado a ser buscado, montando um plano de ação específico para esse fim.

Do: Fase de execução. Nessa fase o plano de ação é colocado em prática e também à prova, já que o ciclo irá gerar um resultado a ser analisado para chegar na melhoria. É importante seguir o planejamento da primeira etapa para que os resultados finais estejam corretos;

Check: Nessa fase é a hora de analisar os resultados obtidos até o momento. Com base nos indicadores, é feita uma análise de como foi a implementação do planejamento, verificando se os resultados foram bons ou ruins. Nesta fase o planejamento será reajustado com os pontos principais da análise.

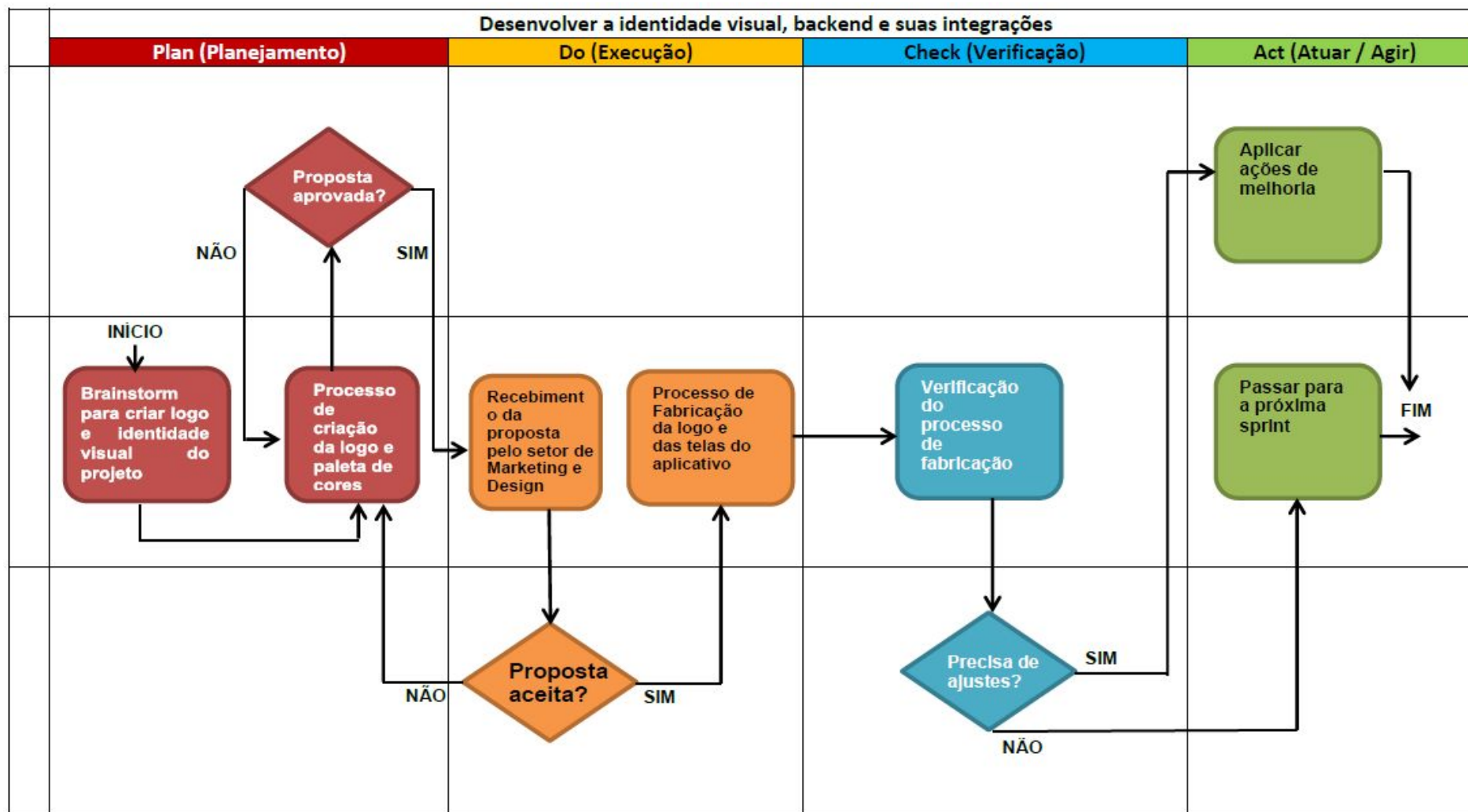
Act: Depois de feita a análise, colocamos em prática os ajustes. Aqui, os erros e problemas deverão ser corrigidos a fim de gerar a melhoria contínua do processo.

Abaixo uma figura representando as principais atividades do ciclo PDCA:

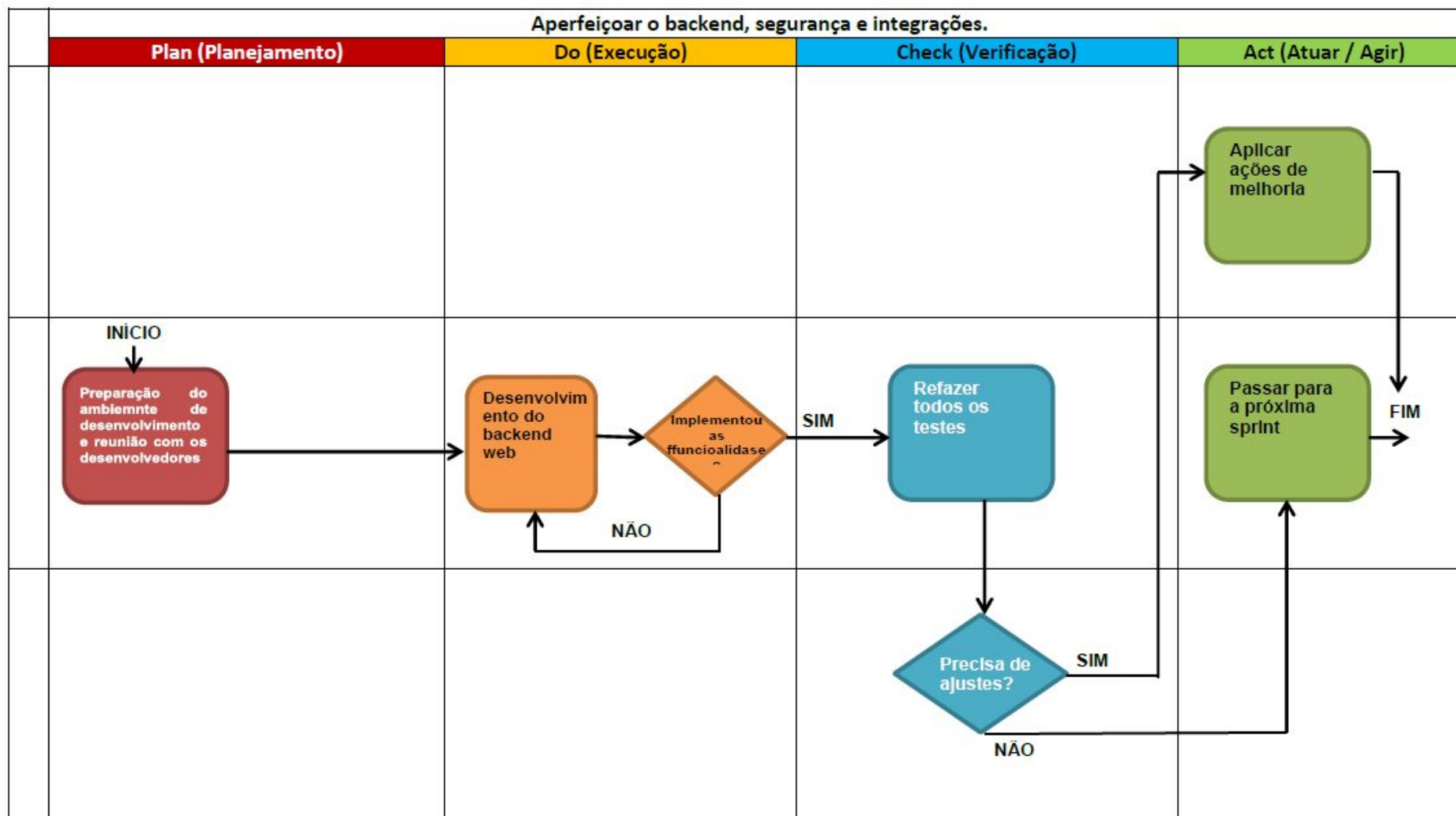


Por ser uma ferramenta fácil e bastante intuitiva, o Ciclo PDCA pode ser aplicado a praticamente qualquer tipo de projeto, dos simples aos mais complexos, é um método que ajuda no direcionamento da equipe para o desenvolvimento de melhorias contínuas, faz com que se tenha mais atenção a identificação de falhas e oportunidades de aprimoramento, além de contribuir para que todos os envolvidos visualizem as mudanças realizadas. Os fluxogramas a seguir foram criados baseados em cada sprint.

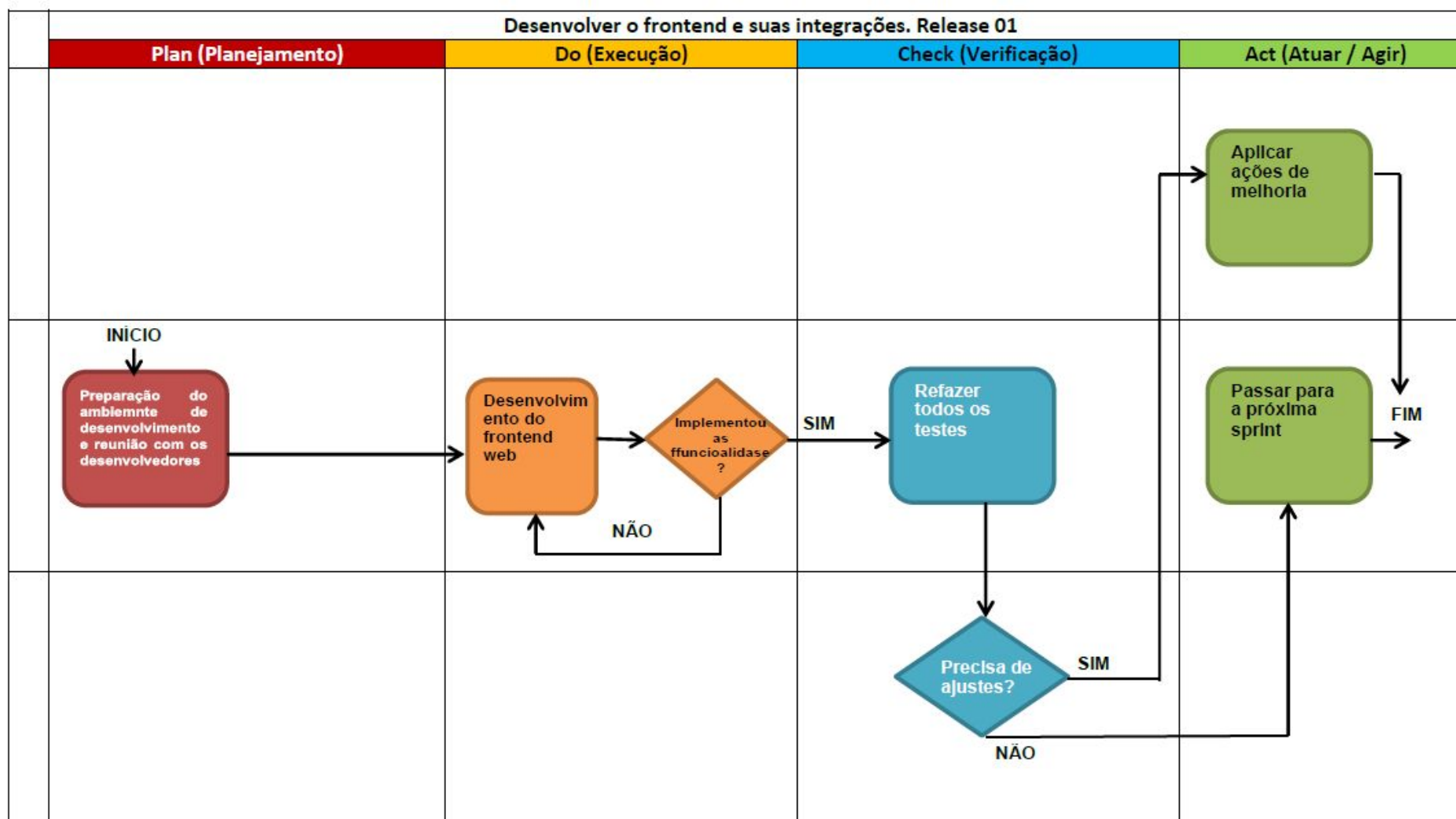
- 1. Sprint 01:** Garantir que a proposta de escolha da identidade visual do aplicativo, escolhendo a paleta de cores, criação da logo e layout das telas da aplicação fossem feitas o mais breve possível e com uma boa aceitação pela equipe. O objetivo foi garantir um layout limpo e agradável para o usuário.



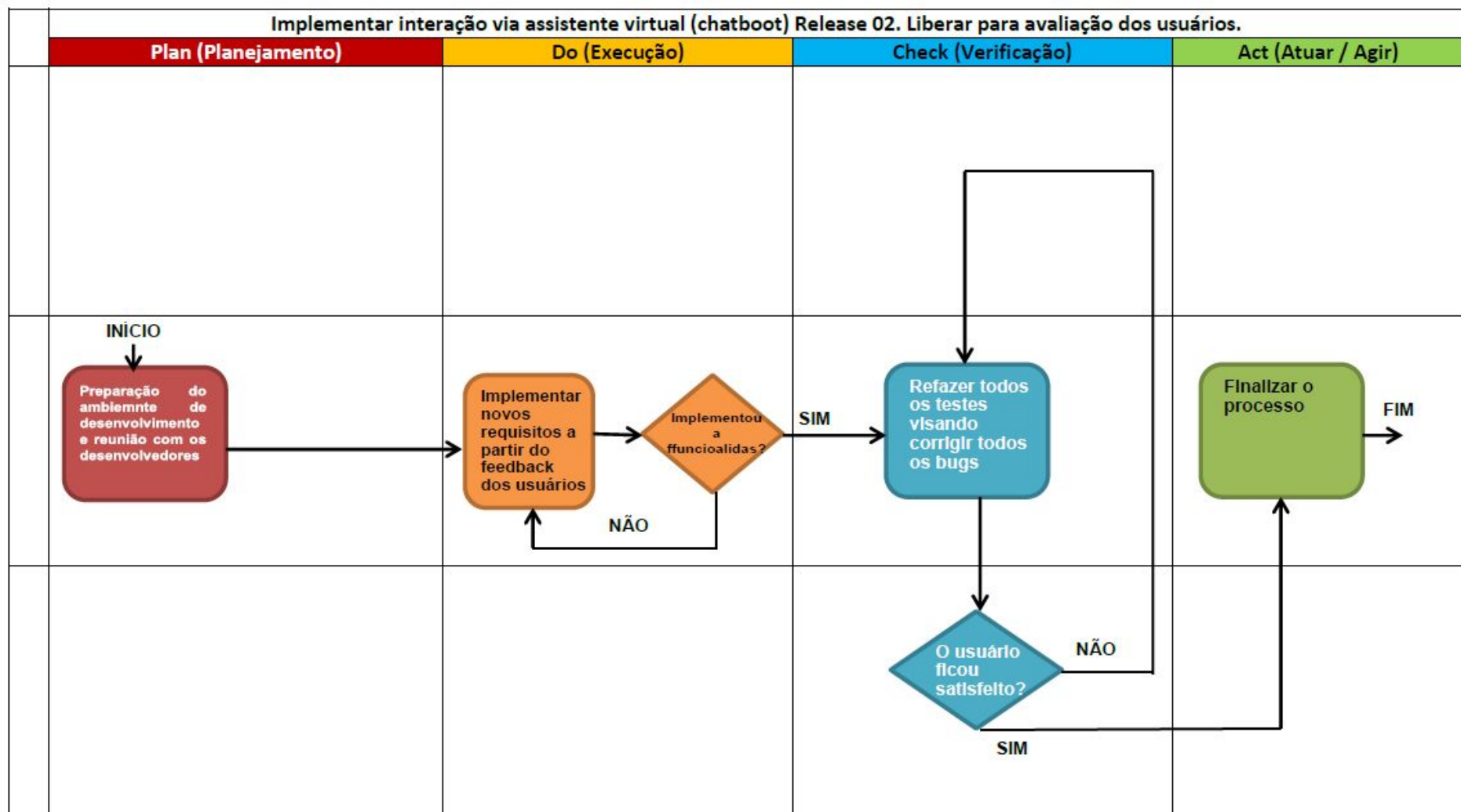
2. **Sprint 02:** Garantir o funcionamento do ambiente de desenvolvimento para o início do backend, buscando sempre realizar testes para cada nova funcionalidade implementado. O objetivo foi realizar checklists para evitar o máximo de erros, visando sempre catalogar os bugs encontrados.



3. **Sprint 03:** Semelhante ao processo aplicado sprint 02, como forma de garantir a qualidade na sprint 03, buscamos corrigir problemas apresentados no ambiente de desenvolvimento para garantir o funcionamento na fase de início do desenvolvimento do frontend e realizar o primeiro release para os usuários, os testes continuaram sendo realizados para cada nova funcionalidade implementada. Visando continuar a evitar erros e catalogar os bugs encontrados.



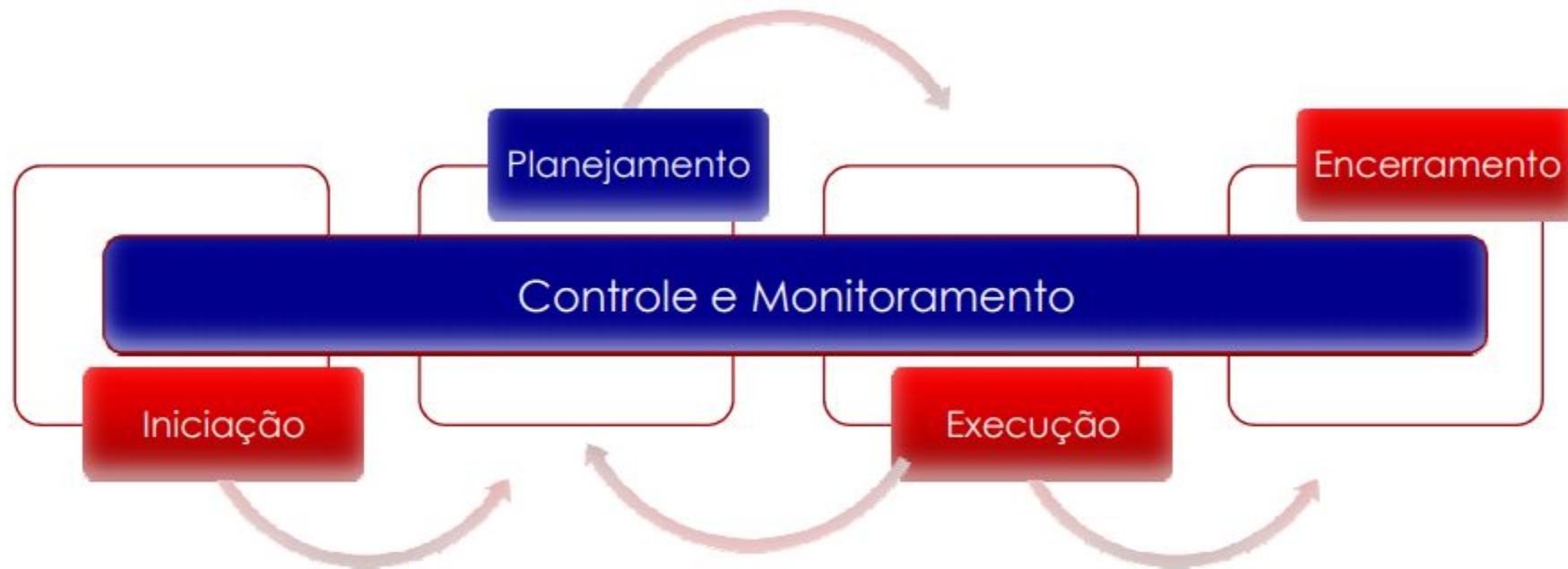
4. **Sprint 04:** Com a fase de implantação do chatbot, a funcionalidade diferenciada do aplicativo My Flow, o objetivo da equipe foi garantir com que os a experiência com o usuário fosse a mais satisfatória possível, foi onde dedicamos um esforço maior em realizar testes para todas as possíveis formas que usuário poderia interagir, buscando identificar e catalogar o maior número possível de bugs que pudesse vir a acontecer.



Concluindo, a escolha dessa aplicação teve como uma meta a melhoria contínua otimização na execução dos processos, possibilitando a redução de custos e o aumento da produtividade. Mesmo os processos sendo considerados satisfatórios, não significam que não sejam passíveis de melhorias. A introdução de novas funcionalidades em cada Sprint, bem como a realização dos checklists, testes e reuniões durante a fase de desenvolvimento, buscavam conseguir melhorias gradativas e contínuas aos processos, agregar maior valor aos resultados do projeto e a garantir uma maior satisfação dos clientes.

2.7 - Gestão dos riscos

Como forma de mitigar os riscos negativos advindos dos possíveis problemas que possam surgir, bem como melhorar os riscos positivos durante toda a fase do projeto, adotamos um processo de gestão de risco baseado no PMBok com controle e monitoramento durante todas as fases do projeto.



1. Planejamento e Gerenciamento de Riscos;

Daremos início por meio de reunião com toda a equipe envolvida no projeto, para definir o processo de gestão de riscos que iremos aplicar. Definindo as atividades, os papéis dos membros, documentação, orçamento, estrutura organizacional, ferramentas e aprovações.

2. Identificar os riscos;

Utilizaremos o EAR - Estrutura Analítica dos Riscos, onde iremos subdividir os riscos por grupo/área, de maneira que possamos ter uma visão mais clara e apoiar as tomadas de decisões de modo mais efetivo.

Risco Técnico: Iremos avaliar a tecnologia utilizada, o conhecimento do técnico do pessoal responsável pelo desenvolvimento da aplicação e todos os demais riscos provenientes do dia-a-dia da implementação do produto.

Risco de Gerenciamento: São as falhas cometidas pelo gerente do projeto, referente ao planejamento, monitoramento, controle e comunicação no projeto.

Risco Comercial: Relacionado ao orçamento do projeto, investidores.

Risco Externo: Legislação, disponibilidade da aplicação no servidor de hospedagem, reclamações/pressão dos clientes.



(Guia PMBOK, 2017, 6ª edição)

Identificação de Riscos da aplicação My Flow:

Nº	Risco Técnico	Risco de Gerenciamento	Risco Comercial	Risco Externo
1	Falta de Conhecimento na tecnologia escolhida	Perca de prazo devido a erros no gerenciamento.	Falta de interesse por parte dos investidores	Alta Concorrência / Mercado Saturado
2	Equipamentos com defeito	Falta de Comunicação efetiva e clara	Orçamento curto para refatoração em caso de falhas ou atualizações	Baixa adesão de usuários
3	Desenvolvedores pode ficar doentes ou indisponíveis para trabalho	Alterações constantes no escopo do projeto		Evasão de Clientes por falta de interesse na plataforma
4	Trabalho excessivo, mais de 40h por semana	Gerente pode ficar doente ou indisponível para trabalho		Indisponibilidade da servidor de Hospedagem
5	Falta de comprometimento de algum membro da equipe	Perca do prazo devido necessidades tecnológicas desconsideradas durante o planejamento das etapas		Mudanças que impactem no modo como as receitas advindas do anúncios são geradas
6				
7				
8				
9				
10				

Algumas perguntas que podemos utilizar em cada grupo de risco acima:

- O que pode acontecer e por que?
- Quais as consequências?
- Qual a probabilidade de ocorrer?
- Até onde o risco é aceitável ou tolerável?

3. Análise qualitativa de riscos;

Qualitativamente iremos, classificar os riscos em categorias, para então saber quais devemos priorizar a tratativa, analisando suas causas e tomando as melhores ações para explorá-los ou mitigar os mesmos.

4. Análise quantitativa de riscos;

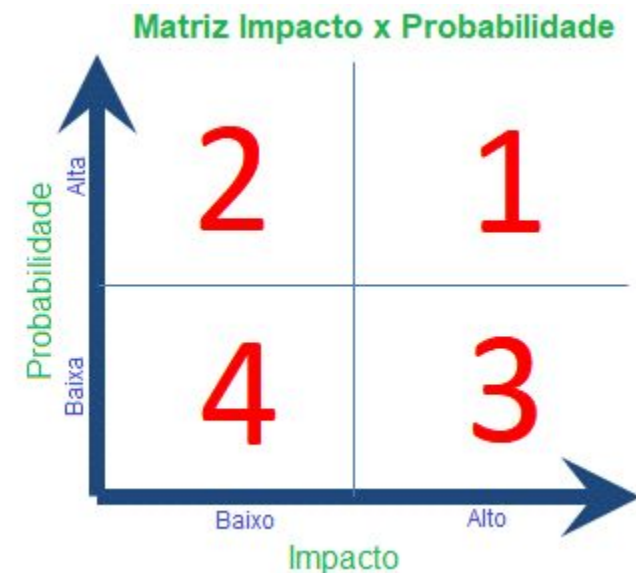
Quantitativamente, iremos avaliar a probabilidade em que os riscos ocorrerem, nos apoiando no impacto que os riscos podem causar ao projeto como um todo.

Quadrante 1 - Alto impacto e Alta probabilidade: Requer ações de mitigação obrigatória e tem prioridade alta.

Quadrante 2 - Baixo impacto e Alta probabilidade: Geralmente são riscos diários e devem ser monitorados constantemente.

Quadrante 3 - Alto impacto e Baixa probabilidade: Normalmente são riscos externos, que fogem do controle da gestão.

Quadrante 4 - Baixo impacto e Baixa probabilidade: São os riscos aceitáveis, devem ser monitorados, porém sem tanta frequência.



4. Resposta aos riscos;

Sabendo que não é possível prever tudo o que possa ocorrer durante a execução do projeto, iremos adotar como prática o envolvimento do cliente, de maneira que o mesmo entenda como é fundamental a sua participação e colaboração no projeto.

Fornecendo pequenas entregas e recebendo o seu feedback, como também avaliando e executando novas atividades que porventura venham a surgir.

Partindo deste pressuposto, elaboramos a nossa proposta de prevenção/mitigação de riscos do projeto My Flow, onde para cada risco identificado na tabela de riscos, teremos uma resposta/ação a ser realizada. Segue nossa proposta:

Nº	Ações Preventivas e/ou Mitigadoras			
1	Realização de Cursos, palestras e treinamentos.	Controle mais rígido em relação ao prazo. Cobrar entregas periódica de tarefas pela gerência.	Buscar constante por novos investidores, patrocinadores e outro incentivos.	Realizar divulgação por meio demídias sociais,mostrando os benefícios da nossa plataforma e focar em diferentes perfils de usuários.
2	Substituição / reparo do equipamentos com defeito.	Realizar reuniões recreativas para integrar a equipe, de maneira a melhorar o relacionamento interpessoal.	Realizar captação de recursos por meio de anúncios, propagrandas, investidores e patrocinadores.	Realizar divulgação por meio demídias sociais,mostrando os benefícios da nossa plataforma e focar em diferentes perfils de usuários.
3	Realocação da tarefa ou redivisão da tarefa entre a equipe.	Verificação e validação de viabilidade. Evitando alterações desnecessárias.	Convidar o cliente a participar de maneira mais efetiva na construção de nova funcionalidades e/ou atualizações	Realizar divulgação por meio demídias sociais,mostrando os benefícios da nossa plataforma e focar em diferentes perfils de usuários.
4	Controle do horário dos trabalhadores, de maneira a evitar horas extras.	Treinar membro para assumir em caso de ausência ou afastamento do Gerente de Projeto.		Viabilizar meios de contingências para hospedagem.
5	Nova conversa para re-orientação ou ajuda ao membro em questão.	Realizar estudo e análise das melhores tecnologias e práticas adotadas pelo mercado e tendências futuras.		Estudo contínuo de novas possibilidades de recursos financeiros.

5. Controle dos riscos.

Monitorar e controlar todos passos anteriores e repetí-los, identificando o surgimento de novos riscos, se riscos antigos foram sanados, mitigados ou explorados, como também se novos surgiram.

3 - PLANEJAMENTO DE EXECUÇÃO DO PROJETO

Apresentamos agora como se dará a sequência de atividades que vão compor todo o desenrolar do projeto.

Conceituamos de semana zero, a semana anterior ao início das Sprints, oportunidade essa utilizada para desenvolvimento e aperfeiçoamento do Business Model Canvas, Project Model Canvas, bem como a realização da reunião de planning poker com o objetivo de estimar os story points necessários para o desenvolvimento do produto.

Nesse momento foram estimados um total de 05 Sprints de 02 semanas cada, para o desenvolvimento do MVP.

3.1 - Sprint 01

Objetivo macro da Sprint	Desenvolver a identidade visual, backend e suas integrações					
Andamento do projeto						0% já concluído
Backlog da Sprint				Responsável pela implementação		
Desenvolver paleta de cores				Matheus		
Criar logo						
Desenvolver imagem das telas e interações						
Implementar CRUDs: <ul style="list-style-type: none">- Implementar lógica do CRUD das tarefas;- Implementar CRUD de cadastro de usuários;- Implementar Banco persistência no Banco de dados;				Flavio e Rodrigo		

3.2 - Sprint 02

Objetivo macro da Sprint	Aperfeiçoar o backend, segurança e integrações.					
Andamento do projeto						20% já concluído

Backlog da Sprint	Responsável pela implementação
<ul style="list-style-type: none"> - Integração de notificação por email; - Funcionalidade push notification em WEB App; - Implementar lógica para funcionalidade de resumo de tarefas; - Implementar lógica para funcionalidade de reordenação; 	Flavio, Matheus e Rodrigo

3.3 - Sprint 03

Objetivo macro da Sprint	Desenvolver o frontend e suas integrações. Release 01					
Andamento do projeto						40% já concluído
Backlog da Sprint	Responsável pela implementação					
<ul style="list-style-type: none"> - Implementar interface front-end seguindo o padrão de WEB APP; - Funcionalidade reordenação de tarefas via segurar e arrastar; - Padrões de acessibilidade; - Botão ou área reservada para o resumo das tarefas; 	Flavio, Matheus e Rodrigo					

3.4 - Sprint 04

Objetivo macro da Sprint	Implementar interação via assistente virtual (chatboot) Release 02. Liberar para avaliação dos usuários.					
Andamento do projeto						60% já concluído

Backlog da Sprint	Responsável pela implementação
o app deverá ter seu fluxo de funcionamento por meio de assistente virtual (chatbot): <ul style="list-style-type: none"> - Integração de api para Interação com chatbot; 	Flavio, Matheus e Rodrigo

Liberar versão beta em plataforma gratuita (Heroku).

3.5 - Sprint 05

Objetivo macro da Sprint	Destinada a cobrir novos requisitos, correções e melhorias que por ventura surgirem a partir dos feedbacks de usuários colhidos na interação do MVP com o cliente final.					
Andamento do projeto						80% já concluído
Backlog da Sprint	Responsável pela implementação					
Nova funcionalidade	Flavio, Matheus e Rodrigo					
Correção de bugs						
Melhoria ...						

Ao final desta Sprint, estaremos com 100% do projeto concluído.

3.6 - Lançamento do MVP

Para finalizar, trazemos por sequência os custos finais da aplicação, com uma estimativa geral, seguida por um custo aproximado por usuário ativo na plataforma.

A primeira parte a ser tratada é o marketing do aplicativo, pensado inicialmente para ser em redes sociais, Facebook e Instagram, e em pesquisas relacionadas na página da plataforma Google. Segundo o Google, cada clique em sua plataforma custa em média R\$0,30, já para o Facebook, custa R\$ 0,30 a R\$ 0,50 (LinksExpert) e no Instagram de R\$ 0,20 e R\$ 1,35 (RocketContent).

Já o custo de infraestrutura para o armazenamento de arquivos estáticos foi decidido pelo uso do Amazon Web Services (AWS). O plano escolhido inicialmente foi o S3 Standard, que é usado para armazenamento de uso geral para qualquer tipo de dados, usado normalmente para dados acessados com frequência. Segundo a AWS Simple Monthly Calculator, cada 1GB de Armazenamento (U\$0,03), 20.000 requisições do tipo GET(U\$0,01) e 2.000(U\$0,01) requisições do tipo PUT/COPY/POST/LIST saem a U\$0,05 por mês.

Para a hospedagem da parte web, admin e arquivos não estáticos, decidimos somente trocar o plano do Heroku para o *Production*, que custa aproximadamente R\$140,00 (U\$25,00) que é um plano mais profissional para aplicativos em produção.

Levando em consideração que cada usuário use 1GB e faça todas essas requisições, o que parece ser bem coerente, cada usuário sairia a R\$0,27 (dólar a 5,38) somando a cada clique gerados pelo marketing R\$1,40 (Google R\$0,30, Facebook R\$0,40 e Instagram R\$0,70) então podemos concluir que cada usuário sairia a R\$1,67.

Por fim, segundo a ThinkMobile, o custo de manutenção do aplicativo é de 20% (R\$2.100,00 no nosso caso). O custo total da produção do mesmo, então, além do custo por usuário, deveremos levar isso em consideração, podendo ser planejado em trabalhos futuros.

Portanto, fazendo um cálculo bruto, podemos estimar que os usuários pagos do MyFlow precisariam investir no mínimo R\$6,00, o que achamos justo, para tirarmos os custos do projeto, além de usuário “se pagar” e ainda termos algum lucro. Também precisaríamos de no mínimo 6.000 usuários pagos para manter esse valor ($12.740 [\text{valor total do PMC} + \text{valor de manutenção} + \text{valor da hospedagem}] / 6.000 = 2,12 // 2,12 + 1,67 = 3,79$), ou seja, nosso lucro seria pelo menos de 45% por usuário.

Referências:

Gestão de Riscos:

<https://www.devmedia.com.br/gerencia-de-riscos-em-desenvolvimento-de-software/28506>;

<https://nerus.com.br/blog/processos/exemplos-riscos-projeto-software/>;

<https://blog.teclogica.com.br/riscos-em-um-projeto-de-software/>;

<https://blog.teclogica.com.br/mitigacao-dos-riscos-em-ti/>;

<https://www.glicfas.com.br/mapeamento-de-riscos/>;

Venâncio, Júlio. Gestão de Riscos em Projetos de Software, UFPE. 2010. Disponível em: <https://www.cin.ufpe.br/~processos/TAES3/slides-2010.2/Apresentacao-Riscos.pdf> Acessado em: 18/11/2020.

A Guide to the Project Management Body of Knowledge (PMBOK Guide), 35 A Guide to the Project Management Body of Knowledge (PMBOK Guide), PMI, 2004;