

Para o desenvolvimento de um projeto bom, que cumpre seu propósito e é desenvolvido de forma eficaz e otimizado é necessário um bom planejamento. Para o nosso software, foram pensados nas seguintes etapas e nos seguintes prazos:

1. **Planejamento e Levantamento de Requisitos – 2 semanas:** entraremos em contato com o consumidor final de nosso software para verificarmos suas necessidades, e assim, planejar os próximos passos do desenvolvimento.
2. **Análise dos requisitos e definição de arquitetura – 1 semana:** análise e definição das tecnologias que melhor atendem os requisitos levantados e elaboração de diagramas de funcionamento do sistema.
3. **Desenho do Sistema – 3 semanas:** definição de design do software e experiência do usuário, criação de *wireframes* e protótipos de telas.
4. **Desenvolvimento – 6 semanas:** codificação do sistema, integração das tecnologias escolhidas, implementação de interface e lógica de negócios.
5. **Testes – 3 semanas:** execução de testes unitários, de integração, de sistema e de aceitação. Corrigir falhas e refinar o código. É feito durante o desenvolvimento.
6. **Implantação – 2 semanas:** preparação do ambiente de produção, implantação automatizada ou manual, migração de dados e configurações.
7. **Monitoramento e manutenção – contínua:** monitoramento do sistema, correção de erros, atualizações de segurança, ajustes de performance, introdução de novas funcionalidades.
8. **Evolução e aperfeiçoamento – contínua:** implementação de novos recursos e melhorias baseadas no feedback dos usuários.
9. **Documentação – 1 semana:** descrição detalhada de classes e funções do sistema e descrição do funcionamento do sistema.

É importante termos ciência de todos os riscos que permeiam o desenvolvimento do projeto, para isso vou listar alguns:

1. **Poucos membros de equipe:** o número baixo de membros na equipe pode gerar atrasos no cronograma e o acúmulo excessivo de tarefas.
2. **Atrasos no cronograma:** o cronograma pode acabar atrasando por conta de diversos fatores, entre eles a equipe enxuta.
3. **Mudança nos requisitos:** por mais que o projeto seja desenvolvido em metodologias ágeis, a mudança de muitos requisitos pode acarretar o atraso do cronograma e pode alterar a verba necessária.
4. **Falta de recursos:** por se tratar do processamento de muitos dados, é possível que o equipamento dos membros da equipe seja um gargalo do projeto, ou até mesmo o custo de serviços para processamento de dados.

O desenvolvimento do software carrega alguns custos, tais quais:

Componente	Custo Estimado (Mensal)
------------	-------------------------

Salário dos desenvolvedores	R\$ 12.000,00
Assinatura AWS (Hospedagem de API)	R\$ 500,00
Processamento em Nuvem (AWS EC2)	R\$ 500,00
<b>Total</b>	<b>R\$ 13.000,00</b>

Para mitigar esses riscos, é importante a definição de métricas para monitoramento e melhoria do software, são eles:

1. **Quantidade de ocorrências de problemas:** para isso, será feito um estudo para saber a quantidade de problemas que o sistema gera, e quais são os principais problemas, a fim de corrigi-los.
2. **Tempo médio para correção dos problemas:** calcular quanto tempo é necessário para resolver determinados problemas, a fim de otimizar as atividades e descobrir possíveis motivos para eventuais atrasos de correções.
3. **Correção e otimização de código:** monitoramento e utilização do SonarQube para revisão dos códigos e sua otimização.