

Lista de Verificação da Qualidade - Processos de Gerenciamento de Projeto do Planejamento (Metodologia da Escritório de Projetos)

Processo	Checklist	Sim	Não	Comentários
1. ehSoja	Aplicativo com todas as funcionalidades (novas e anteriores) estáveis e devidamente testadas	x		
1.1. Definição dos requisitos	Saber quais são todas as necessidades do produto a ser desenvolvido	x		
1.1.1. Tirar dúvidas sobre o projeto;	Ter em mente boa parte dos detalhes que precisam ser considerados durante o desenvolvimento da solução	x		
1.1.2. Analisar o conhecimento de cada integrante da equipe	Registrar em quais requisitos cada um pode ajudar da melhor forma	x		
1.1.3. Ajudar a chegar na solução estipulada primeiramente	Assim que todo o backlog estiver definido e priorizado.	x		
1.2. Gerenciamento do projeto	Saber qual é a estimativa de tempo e custo para o desenvolvimento do projeto	x		
1.2.1. Definição do PO e do SM	Somente quando o <i>Scrum Master</i> e o <i>Product Owner</i> forem escolhidos	x		
1.2.2. Criação do Termo de Abertura do Projeto	Somente quando o documento estiver inteiramente preenchido e o professor aprovar	x		
1.2.3. Criação da Declaração de Escopo do Projeto	Somente quando o documento estiver inteiramente preenchido e o professor aprovar	x		
1.2.4. Criação do gráfico de Burndown	Somente quando o documento estiver com o seu <i>template</i> criado e os cálculos forem feitos de forma automática	x		
1.3. Desenvolvimento	Todos os desenvolvedores precisam ter os projetos em seu computador e, também, conseguir executá-los	x		
1.3.1. Sprint 1	Conseguir identificar a planta de soja e vagens em uma imagem	x		
1.3.1.1. Treinamento do modelo básico para reconhecer as plantas de soja	Reconhecer a soja na imagem	x		

1.3.1.1.1 Estudos sobre reconhecimento de imagem	Saber como fazer o treinamento do modelo para localização das plantas/vagens	x		
1.3.1.1.2. Tratamento das imagens recebidas	Assim que todas as imagens de treinamento estiverem no mesmo padrão e de acordo com o que a entrada do modelo pré-treinado aceita	x		
1.3.1.1.3. Criação do Dataset para o treinamento do modelo de reconhecimento de plantas de soja	Dividir quais imagens serão utilizadas para treinamento do modelo e quais serão utilizadas para teste. Realizar a anotação das vagens em todas as imagens	x		
1.3.1.1.4 Treinamento de um modelo que reconheça plantas de soja	Utilizar as imagens padronizadas e as suas anotações para gerar um modelo de reconhecimento de plantas de soja	x		
1.3.1.2.1. Treinamento do modelo básico para reconhecer as vagens na planta de soja	Aprimorar este modelo para que ele reconheça vagens também	x		
1.3.1.2.2. Estudos sobre reconhecimento de imagem	Saber como fazer o treinamento do modelo para circular as plantas/vagens	x		
1.3.1.2.3. Estudos sobre segmentação de imagem	Saber como fazer o treinamento do modelo para circular as plantas/vagens	x		
1.3.1.2.4. Criação do Dataset para o treinamento do modelo de reconhecimento de vagens	Dividir quais imagens serão utilizadas para treinamento do modelo e quais serão utilizadas para teste. Realizar a anotação das vagens em todas as imagens	x		
1.3.1.2.5. Treinamento do modelo que reconheça as vagens na planta de soja	Utilizar as imagens padronizadas e as suas anotações para gerar um modelo de reconhecimento de vagens	x		
1.3.1.3.1. Destacar, na imagem, a planta de soja e as suas vagens	Com os resultados em mãos, adicionar os dados na imagem original	x		
1.3.1.3.2. Obter imagens com as suas características reconhecidas e os dados do reconhecimento	Obter a imagem original e as suas características reconhecidas	x		

1.3.1.3.3. Realizar a interpretação dos dados do reconhecimento das imagens	Processar os dados recebidos pelo reconhecimento	x		
1.3.1.3.4. Exibir os dados do reconhecimento das imagens juntamente à imagem em si	Obter a imagem original juntamente às suas características reconhecidas	x		
1.3.2.1.1. Sprint 2	Exibir corretamente no aplicativo os dados reconhecidos a partir do envio da foto de uma planta de soja	x		
1.3.2.1.1. Criação/alteração da interface do aplicativo onde as imagens das novas plantas são registradas como amostras	A interface deve ser completamente funcional e intuitiva para o usuário	x		
1.3.2.1.1.1. Conseguir executar o servidor do aplicativo eSoja	Executar o servidor do aplicativo sem nenhum erro			
1.3.2.1.1.1.1. Reconfigurar aplicativo para o funcionamento de todas as suas áreas	Reconfigurar chaves de API e IPs para o correto funcionamento do app	x		
1.3.2.1.1.1.2. Realizar o upload do código do servidor com as devidas correções	Enviar o código ao repositório	x		
1.3.2.1.1.1.3. Criar readme contendo instruções para montagem do ambiente de desenvolvimento e de instalação do servidor para a equipe	Escrever todas as instruções para executar o servidor	x		
1.3.2.1.2.1. Conseguir executar e instalar o aplicativo eSoja no celular	Executar o app do aplicativo sem nenhum erro em um dispositivo móvel	x		

1.3.2.1.2.1.1. Realizar upload do código do aplicativo contendo correções	Reconfigurar chaves de API e IPs para o correto funcionamento do app	x		
1.3.2.1.2.1.2. Criar readme contendo instruções para montagem do ambiente de desenvolvimento e de instalação do aplicativo para a equipe	Escrever todas as instruções para executar o servidor	x		
1.3.2.1.2. Interface para exibir ao usuário as amostras analisadas	A interface deve ser completamente funcional, intuitiva para o usuário e deve exibir a identificação da planta e suas vagens corretamente	x		
1.3.2.1.2.1. Desenvolver serviço que repassa as imagens recebidas ao modelo de reconhecimento e as retorna com os dados prontos para enviar ao aplicativo	O serviço, no servidor, deve receber uma imagem em base64 e enviá-la ao Tensorflow para o reconhecimento das suas características	x		
1.3.2.1.2.2. Desenvolver controlador para receber as requisições do aplicativo e processar as imagens recebidas	O controlador deve disponibilizar uma rota para que o serviço de reconhecimento possa ser acessado	x		
1.3.2.1.2.3. Desenvolver interface, no aplicativo eSoja, para o envio das amostras que serão utilizadas para o reconhecimento	A interface deve ser completamente funcional, intuitiva para o usuário e deve exibir a identificação da planta e suas vagens corretamente	x		
1.3.2.1.2.4. Desenvolver interface, no aplicativo eSoja, para a exibição dos resultados do reconhecimento das imagens enviadas	A interface deve ser completamente funcional, intuitiva para o usuário e deve exibir a identificação da planta e suas vagens corretamente	x		

1.3.3. Sprint 3	Exibir corretamente no aplicativo os dados reconhecidos, com maior qualidade, a partir do envio da foto de uma planta de soja		x	Sprint ainda não iniciada.
1.3.3.1. Aperfeiçoar a precisão do modelo de reconhecimento de plantas de soja e vagens	Otimizar o código para que haja maior acurácia e os resultados sejam exibidos rapidamente ao usuário		x	Sprint ainda não iniciada.
1.3.3.1.1. Realizar estudos sobre como o modelo de reconhecimento pode melhorar	Consultar materiais na internet e obter pelo menos uma técnica que possa melhorar ainda mais a acurácia do reconhecimento		x	Sprint ainda não iniciada.
1.3.3.1.2. Desenvolver melhorias no modelo de reconhecimento	Com as técnicas pesquisadas, implementá-las		x	Sprint ainda não iniciada.
1.3.3.1.3. Treinar um novo modelo contendo as melhorias	Utilizar o novo código, com as melhorias, para treinar um novo modelo que será utilizado nas novas previsões		x	Sprint ainda não iniciada.
1.3.3.1.4. Realizar o upload do novo modelo para que ele possa ser utilizado	Realizar o upload do novo modelo no repositório		x	Sprint ainda não iniciada.
1.3.3.1.5. Substituir o modelo antigo pelo mais novo e mais eficiente	Alterar o serviço de reconhecimento de imagens, no servidor, para utilizar o novo modelo		x	Sprint ainda não iniciada.
1.3.3.2.1. Realizar a contagem de quantas vagens foram encontradas em uma planta de soja	A contagem feita pelo código desenvolvido deve passar em todos os testes com diversas imagens		x	Sprint ainda não iniciada.
1.3.3.2.1.1. Desenvolver código para registrar a quantidade de vagens foram encontradas na planta de soja	Desenvolver código para registrar no banco de dados as informações obtidas no reconhecimento		x	Sprint ainda não iniciada.
1.3.3.2.1.2. Inserir informação na resposta a ser enviada ao aplicativo eSoja	Alterar o código para enviar os dados obtidos para o aplicativo		x	Sprint ainda não iniciada.

1.3.3.1.2. Exibir, no aplicativo eSoja, a quantidade de vagens, de acordo com o resultado da análise da imagem	Criação de uma interface que o usuário possa visualizar o reconhecimento realizado		x	Sprint ainda não iniciada.
1.3.3.1.2.1. Criar uma interface, no aplicativo eSoja, que exiba a quantidade de vagens reconhecidas	Criação de uma interface que o usuário possa visualizar os valores obtidos através do reconhecimento realizado		x	Sprint ainda não iniciada.
1.3.4. Sprint 4	Exibir correta e rapidamente no aplicativo os dados reconhecidos a partir do envio da foto de uma planta de soja		x	Sprint ainda não iniciada.
1.3.4.1. Exibir a quantidade de grãos nas amostras, de acordo com o resultado da análise da imagem	Exibir na interface do app a quantidade de grãos nas plantas de soja		x	Sprint ainda não iniciada.
1.3.4.1.1. Estudar sobre cálculo da estimativa de grãos na planta	Registrar, em um documento dentro do repositório, qual será o cálculo que precisa ser feito		x	Sprint ainda não iniciada.
1.3.4.1.2. Desenvolver cálculo da estimativa de grãos na planta	Consultar documento com o cálculo e transcrevê-lo no código do nosso servidor		x	Sprint ainda não iniciada.
1.3.4.1.3. Inserir informação na resposta a ser enviada ao aplicativo eSoja	Inserir informação de quantidade de sementes na resposta a ser enviada ao aplicativo		x	Sprint ainda não iniciada.
1.3.4.1.4. Desenvolver alteração, na interface do aplicativo eSoja, que irá exibir a estimativa de grãos na planta	Exibir no aplicativo todas as informações obtidas através da análise da imagem enviada pelo usuário		x	Sprint ainda não iniciada.
1.3.4.1.5. Realizar melhorias finais no aplicativo	Melhorar a experiência do usuário do aplicativo		x	Sprint ainda não iniciada.

1.4. Testes	O aplicativo deve possuir a maior quantidade de testes possíveis, que cubram todos os cenários imagináveis e erros previsíveis		x	O projeto ainda está em andamento.
1.4.1. Desenvolvimento de testes unitários	Os testes devem passar em todos os casos desenvolvidos		x	O projeto ainda está em andamento.
1.4.2. Realização de testes funcionais	As principais funcionalidades não podem “quebrar”		x	O projeto ainda está em andamento.
1.4.3. Realização de simulações envolvendo os mais diferentes cenários	O aplicativo como um todo utilizando nossas alterações precisa permanecer consistente		x	O projeto ainda está em andamento.
1.5. Entrega do projeto	As funcionalidades preexistentes do aplicativo devem permanecer funcionando e o novo módulo para o reconhecimento de imagens também		x	Projeto ainda não finalizado.
1.5.1. Gravação de uma simulação do software em funcionamento	O vídeo deve conter todas as alterações que desenvolvemos, de forma a valorizar o nosso esforço		x	Projeto ainda não finalizado.
1.5.2. Apresentação ao cliente	Apresentar a evolução do projeto durante o seu desenvolvimento, mostrando o nosso esforço		x	Projeto ainda não finalizado.