|  |  |  |  |
| --- | --- | --- | --- |
| **Controle de Versões** | | | |
| **Versão** | **Data** | **Autor** | **Notas da Revisão** |
| 1 | 07/09/2022 | Bárbara dos Santos Port | Primeira versão do conteúdo do documento. |

# Objetivos deste documento

Descrever de forma clara qual trabalho deverá ser realizado e quais entregas serão produzidas.

# Objetivos do projeto

O app eSoja possui uma área para estimativa de cultivo e os dados sobre as plantas de soja são inseridos de forma manual, ou seja, pelo próprio usuário (quantidade de vagens, quantidade de grãos etc.). Para isso é necessário certo esforço e pode causar erros. Sendo assim, será desenvolvida uma nova funcionalidade para que tais dados sejam obtidos de forma automática a partir de uma foto da planta.

# Escopo do Produto

Após a conversa com o cliente e a análise das possíveis soluções, foram definidas as seguintes funcionalidades para a melhoria do aplicativo:

* 1ª Sprint:
  + Treinamento do modelo para reconhecer as plantas de soja;
  + Treinamento do modelo para reconhecer as vagens na planta de soja;
  + Marcar na imagem as vagens encontradas na planta de soja;
  + Fazer a contagem de quantas vagens foram encontradas na planta de soja.
* 2ª Sprint:
  + Criação/alteração da interface onde novas plantas são registradas para comportar as novas necessidades;
  + Interface para exibir ao usuário o resultado da imagem analisada, onde as vagens e plantas são marcadas na imagem;
  + Preencher o valor de vagens na planta de acordo com o resultado da análise da imagem.
* 3ª Sprint:
  + Criação/alteração da interface onde novas plantas são registradas para comportar as novas necessidades;
  + Treinamento do modelo para reconhecer vagens com X grãos de soja;
  + Preencher o valor de grãos na planta de acordo com o resultado da análise da imagem.
* 4ª Sprint:
  + Ajustes de performance e qualidade do reconhecimento.

# Exclusões do projeto / Fora do Escopo

Serão consideradas atividades fora do escopo, quaisquer outros requisitos e/ou pedidos que não estejam reportadas no Escopo do Produto, pois todo o escopo foi validado previamente com o cliente e não houve sugestão de mudança.

# Restrições

* É necessário manter todas as tecnologias que foram utilizadas na primeira versão do aplicativo;
* Cada integrante deverá receber R$50,00 por hora, portanto, em conformidade com o tempo estimado:
  + Ao final de uma sprint o custo não poderá exceder R$1.071,00 por integrante;
  + Ao final de uma sprint o custo não poderá exceder R$7.500,00 considerando toda a equipe;
  + O projeto, em sua totalidade, não poderá exceder o custo de R$30.000,00.
* Cada integrante deverá dispor de 1h por dia para desenvolver o projeto, portanto:
  + Ao final de uma sprint cada integrante não poderá ter utilizado mais de 21h;
  + Ao final de uma sprint o tempo total utilizado para o desenvolvimento, tendo em vista toda a equipe, não poderá ser maior que 150h;
  + O projeto, como um todo, não poderá exceder 600h considerando todos os integrantes e todas as sprints.

# Premissas

* Todos os desenvolvedores devem possuir em seus equipamentos de trabalho todas as tecnologias necessárias para a realização do projeto;
* O *Product Owner* deve estar em contato com o cliente e verificar todos os dias a existência de avisos e respostas de perguntas realizadas;
* Cada integrante deverá receber R$50,00 por hora, portanto, em conformidade com o tempo estimado:
  + Ao final de uma sprint o custo deverá ser de R$1.071,00 por integrante;
  + Ao final de uma sprint o custo deverá ser de R$7.500,00 considerando toda a equipe;
  + O projeto, em sua totalidade, deverá custar R$30.000,00.
* Cada integrante deverá dispor de 1h por dia para desenvolver o projeto, portanto:
  + Ao final de uma sprint cada integrante deverá ter reservado 21h;
  + Ao final de uma sprint o tempo total utilizado para o desenvolvimento, tendo em vista toda a equipe, deverá ser de 150h;
  + Ao final do projeto deverão ser gastas 600h considerando todos os integrantes e todas as sprints.
* O aplicativo, após o desenvolvimento da nova funcionalidade, deverá conter, ainda, as mesmas funcionalidades que já existiam anteriormente.

# Estrutura Analítica do Projeto

1. ehSoja
   1. Definição dos requisitos:
      1. Tirar dúvidas sobre o projeto;
      2. Analisar o conhecimento de cada integrante da equipe;
      3. Ajudar a chegar na solução estipulada primeiramente.
   2. Gerenciamento do projeto:
      1. O projeto será gerenciado pela equipe através de 1 (um) *Scrum Master*, 1 (um) *Product Owner* e 5 (cinco) desenvolvedores, além de 5 (cinco) integrantes da empresa contratada que farão o gerenciamento, por serem clientes do certame;
      2. Criação do Termo de Abertura do Projeto;
      3. Criação da Declaração de Escopo do Projeto;
      4. Criação do gráfico de *Burndown*.
   3. Desenvolvimento:
      1. Sprint 0:
         1. Treinamento do modelo para reconhecer as plantas de soja;
         2. Treinamento do modelo para reconhecer as vagens na planta de soja;
         3. Marcar na imagem as vagens encontradas na planta de soja;
         4. Fazer a contagem de quantas vagens foram encontradas na planta de soja.
      2. Sprint 1:
         1. Criação/alteração da interface onde novas plantas são registradas para comportar as novas necessidades;
         2. Interface para exibir ao usuário o resultado da imagem analisada, onde as vagens e plantas são marcadas na imagem;
         3. Preencher o valor de vagens na planta de acordo com o resultado da análise da imagem.
      3. Sprint 2:
         1. Criação/alteração da interface onde novas plantas são registradas para comportar as novas necessidades;
         2. Treinamento do modelo para reconhecer vagens com X grãos de soja;
         3. Preencher o valor de grãos na planta de acordo com o resultado da análise da imagem.
      4. Sprint 3:
         1. Ajustes de performance e qualidade do reconhecimento.
   4. Testes:
      1. Desenvolvimento de testes unitários;
      2. Realização de testes funcionais;
      3. Realização de simulações envolvendo os mais diferentes cenários.
   5. Entrega do projeto:
      1. Gravação de uma demonstração do *software* em funcionamento;
      2. Apresentação ao cliente.

# Critérios de Aceitação

|  |  |  |
| --- | --- | --- |
| **Código EAP** | **Entrega** | **Critérios de aceitação** |
| 1 | ehSoja | Aplicativo |
| 1 | Definição dos requisitos | Saber quais são todas as necessidades do produto a ser desenvolvido |
| 1.1 | Tirar dúvidas sobre o projeto | Ter em mente boa parte dos detalhes que precisam ser considerados durante o desenvolvimento da solução |
| 1.2 | Analisar o conhecimento de cada integrante da equipe | Registrar em quais requisitos cada um pode ajudar da melhor forma |
| 1.3 | Ajudar a chegar na solução estipulada primeiramente | Assim que todo o backlog estiver definido e priorizado. |
| 2 | Gerenciamento do projeto | Saber qual é a estimativa de tempo e custo para o desenvolvimento do projeto |
| 2.1 | O projeto será gerenciado pela equipe através de 1 (um) *Scrum Master*, 1 (um) *Product Owner* e 5 (cinco) desenvolvedores, além de 5 (cinco) integrantes da empresa contratada que farão o gerenciamento, por serem clientes do certame | Somente quando o *Scrum Master* e o *Product Owner* forem escolhidos |
| 2.2 | Criação do Termo de Abertura do Projeto | Somente quando o documento estiver inteiramente preenchido e o professor aprovar |
| 2.3 | Criação da Declaração de Escopo do Projeto | Somente quando o documento estiver inteiramente preenchido e o professor aprovar |
| 2.4 | Criação do gráfico de *Burndown* | Somente quando o documento estiver com o seu *template* criado e os cálculos forem feitos de forma automática |
| 3 | Desenvolvimento | Todos os desenvolvedores precisam ter os projetos em seu computador e, também, conseguir executá-los |
| 3.1 | Sprint 0 | Conseguir identificar a planta de soja e vagens em uma imagem, além de contar a quantidade de vagens |
| 3.1.1 | Treinamento do modelo para reconhecer as plantas de soja | Conseguir reconhecer, em uma imagem, o que é (ou não) uma planta de soja |
| 3.1.2 | Treinamento do modelo para reconhecer as vagens na planta de soja | Conseguir reconhecer vagens em uma planta de soja |
| 3.1.3 | Marcar na imagem as vagens encontradas na planta de soja | A partir das vagens reconhecidas, ter a quantidade delas na planta de soja |
| 3.1.4 | Fazer a contagem de quantas vagens foram encontradas na planta de soja | A contagem feita pelo código desenvolvido deve passar em todos os testes com diversas imagens |
| 3.2 | Sprint 1 | Exibir corretamente no aplicativo os dados reconhecidos a parir do envio da foto de uma planta de soja |
| 3.2.1 | Criação/alteração da interface onde novas plantas são registradas para comportar as novas necessidades | A interface deve ser completamente funcional e intuitiva para o usuário |
| 3.2.2 | Interface para exibir ao usuário o resultado da imagem analisada, onde as vagens e plantas são marcadas na imagem | A interface deve ser completamente funcional, intuitiva para o usuário e deve exibir a identificação da planta e suas vagens corretamente |
| 3.2.3 | Preencher o valor de vagens na planta de acordo com o resultado da análise da imagem | A interface deve ser completamente funcional, intuitiva para o usuário e deve exibir a quantidade de vagens corretamente |
| 3.3 | Sprint 2 | Exibir corretamente no aplicativo os dados reconhecidos a parir do envio da foto de uma planta de soja |
| 3.3.1 | Criação/alteração da interface onde novas plantas são registradas para comportar as novas necessidades | A interface deve ser completamente funcional e intuitiva para o usuário |
| 3.3.2 | Treinamento do modelo para reconhecer vagens com X grãos de soja | Reconhecer corretamente cada grão de soja contido em cada uma das vagens, a partir de uma imagem enviada |
| 3.3.3 | Preencher o valor de grãos na planta de acordo com o resultado da análise da imagem | A interface deve ser completamente funcional, intuitiva para o usuário e deve exibir a quantidade de grãos de soja corretamente |
| 3.4 | Sprint 3 | Exibir correta e rapidamente no aplicativo os dados reconhecidos a parir do envio da foto de uma planta de soja |
| 3.4.1 | Ajustes de performance e qualidade do reconhecimento | Otimizar o código para que os resultados sejam exibidos rapidamente ao usuário |
| 4 | Testes | O aplicativo deve possuir a maior quantidade de testes possíveis, que cubram todos os cenários imagináveis e erros previsíveis |
| 4.1 | Desenvolvimento de testes unitários | Os testes devem passar em todos os casos desenvolvidos |
| 4.2 | Realização de testes funcionais | As principais funcionalidades não podem “quebrar” |
| 4.3 | Realização de simulações envolvendo os mais diferentes cenários | O aplicativo como um todo utilizando nossas alterações precisa permanecer consistente |
| 5 | Entrega do projeto | As funcionalidades preexistentes do aplicativo devem permanecer funcionando e o novo módulo para o reconhecimento de imagens também |
| 5.1 | Gravação de uma demonstração do *software* em funcionamento | O vídeo deve conter todas as alterações que desenvolvemos, de forma a valorizar o nosso esforço |
| 5.2 | Apresentação ao cliente | Apresentar a evolução do projeto durante o seu desenvolvimento, mostrando o nosso esforço |

|  |  |  |
| --- | --- | --- |
| **Aprovações** | | |
| **Participante** | **Assinatura** | **Data** |
| Patrocinador do Projeto |  |  |
| Gerente do Projeto | Bárbara Port | 14/09/2022 |