UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET

LUCAS SAMUEL PEREIRA GODOY

SISTEMA DE APOIO PARA MANUTENÇÕES PREVENTIVAS DE COLHEITADEIRAS E TRATORES AGRÍCOLAS

MONOGRAFIA DE TRABALHO DE CONCLUSÃO DE CURSO DE GRADUAÇÃO

GUARAPUAVA
2021

LUCAS SAMUEL PEREIRA GODOY

SISTEMA DE APOIO PARA MANUTENÇÕES PREVENTIVAS DE COLHEITADEIRAS E TRATORES AGRÍCOLAS

Projeto de Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 1, do Curso Superior de Tecnologia em Sistemas para Internet – TSI – da Universidade Tecnológica Federal do Paraná – UTFPR – Câmpus Guarapuava, como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: Prof. Me. Guilherme da Costa Silva

Coorientador: Prof. Me. Fábio Leandro Janiszevski

GUARAPUAVA
2021

4.0 Internacional

Esta licença permite remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es) e que licenciem as novas criações sob termos idênticos. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

RESUMO

PEREIRA, LUCAS SAMUEL GODOY. Sistema de apoio para manutenções preventivas de colheitadeiras e tratores agrícolas. 2021. 28 f. Monografia de Trabalho de Conclusão de Curso de graduação – Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava, 2021.

Problemas em maquinários agrícolas, em etapas cruciais do plantio à colheita, acarretam em atrasos e prejuízos na produção agrícola. Isto está relacionado diretamente a não realização programada de manutenções preventivas. Desta forma, é de suma importância que as medidas preventivas presentes nos manuais de fábrica sejam executas, prevenindo que o maquinário apresente falhas, mantendo-os em boas condições e assim, prolongando a vida útil do mesmo. Apesar das manutenções já estarem presente nos manuais de fábrica, conhecer as especificações técnicas de cada máquina e lembrar-se de realizar as manutenções preventivas se torna uma tarefa difícil, por exemplo, em uma única propriedade é possível existir inúmeros tratores de diferentes marcas e especificações, cada um possuindo tecnologias diferentes e uma agenda de manutenção própria, exigindo um controle rigoroso dos responsáveis. Tendo em vista o problema encontrado, este trabalho tem por objetivo a implementação de um sistema capaz de instruir o agricultor sobre manutenções preventivas de colheitadeiras e tratores agrícolas, alertando e instruindo sobre manutenções e mantendo um histórico de cada maquinário, auxiliando assim o usuário a se obter um controle de sua frota agrícola.

Palavras-chave: Manutenção preventiva. Maquinários agrícolas. Sistema. Software. Gerenciamento.

ABSTRACT

PEREIRA, LUCAS SAMUEL GODOY. Support system for preventive maintenance of harvesters and agricultural tractors. 2021. 28 f. Monografia de Trabalho de Conclusão de Curso de graduação – Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava, 2021.

Problems in agricultural machinery, at crucial stages from planting to harvest, lead to delays and losses in agricultural production. This is directly related to the scheduled failure to carry out preventive maintenance. In this way, it is of utmost importance that the preventive measures present in the factory manuals are carried out, preventing the machinery from failing, keeping them in good condition and thus extending its useful life. Although maintenance is already present in the factory manuals, knowing the technical specifications of each machine and remembering to carry out preventive maintenance becomes a difficult task, for example, in a single property it is possible to have numerous tractors of different brands and specifications , each with different technologies and its own maintenance schedule, requiring strict control of those responsible. In view of the problem encountered, this work aims to implement a system capable of instructing the farmer on preventive maintenance of harvesters and agricultural tractors, alerting and instructing on maintenance and maintaining a history of each machinery, thus helping the user to get control of your agricultural fleet.

Keywords: Preventive maintenance. Agricultural machinery. System. Software. Management.

LISTA DE FIGURAS

Figura 1 – Tela de cadastro de Planos - Gestão de Frotas. 5
Figura 2 – Tela de resultados do aplicativo Carro 6
Figura 3 – Language Ranking 2020 9
Figura 4 – Most Popular Backend Frameworks 9
Figura 5 – Modelo de arquitetura MVC. 10
Figura 6 – Método Scrum 11
Figura 7 – Padrão de escrita de uma User Story. 13
Figura 8 – Tela de Login. 15
Figura 9 – Tela inicial do sistema. 16
Figura 10 – Tela de cadastro de usuários. 16
Figura 11 – Tela de cadastro de maquinários. 17
Figura 12 – Exemplo QRCODE para identificação do maquinário. 17
Figura 13 – Tela para pesquisa de um maquinário. 18
Figura 14 – Inserção do número do hodômetro. 18
Figura 15 – Salvamento hodômetro. 18
Figura 16 – Tela de seleção de maquinário. 19
Figura 17 – Tela de cadastro de manutenções. 20
Figura 18 – Tela de checagem de manutenções. 21
Figura 19 – Modal com formulário da realização da manutenção. 21
Figura 20 – Modal com formulário de adiamento da manutenção. 22
Figura 21 – Tela de geração do relatório de manutenções. 22
Figura 22 – Exemplo do relatório de manutenções. 23
Figura 23 – Tela de geração do relatório de gastos com manutenções. 23
Figura 24 – Exemplo do relatório de gastos com manutenções. 24
Figura 25 – Diagrama de Entidade Relacionamento. 25

LISTA DE QUADROS

Quadro 1 – Cronograma de Atividades. 3
Quadro 2 – Comparação de funções entre os sistemas. 7
Quadro 3 – Product backlog. 14

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
QRCODE	Código de resposta rápida. (do inglês Quick Response Code)
MVC	Model-View-Controller (em português Modelo, Visão e Controle)
HTML	Linguagem de Marcação de Hipertexto (do inglês Hypertext Markup Language)
PDF	Formato Portátil de Documento (do inglês Portable Document Format)
PHP	Pré-processador de hipertexto (do inglês Hypertext Preprocessor)
SQL	Linguagem estruturada para consultas (do inglês Structured Query Language)
SGBD	Sistema de Gerenciamento de Banco de Dados (do inglês Data Base Management System)

SUMÁRIO

1 – INTRODUÇÃO 1
1.1 OBJETIVOS 2
1.1.1 Objetivo Geral 2
1.1.2 Objetivos Específicos 3
1.2 PLANEJAMENTO DO TRABALHO 3

2 – TRABALHOS CORRELATOS 4
2.1 SISTEMAS SIMILARES 4
2.1.1 Manutenção Tratores Agrícolas 4
2.1.2 Gestão de Frotas. 4
2.1.3 Carro - Manutenção Programada 6
2.2 DIFERENCIAL TECNOLÓGICO 7

3 – REFERENCIAL TEÓRICO 8
3.1 METODOLOGIA DE DESENVOLVIMENTO 12

4 – ANÁLISE E PROJETO DO SISTEMA 13
4.1 LEVANTAMENTO DE REQUISITOS 13
4.2 PROTOTIPAÇÃO DAS TELAS 15
4.3 MODELAGEM DO BANCO 24

5 – CONCLUSÃO 26

Referências 27

1 INTRODUÇÃO

Com o aumento da população e a consequente necessidade de se produzir cada vez mais alimentos, tem se tornado muito necessário levar a modernização ao campo. O governo brasileiro, a partir de 1965, com a criação do Sistema Nacional de Crédito Rural (SNCR), oficializou o financiamento de maquinários e implementos agrícolas, facilitando o acesso dos agricultores aos insumos modernos, gerando maior produtividade, reduzindo os custos e formando um excedente agrícola exportável (VIAN, 2017). Os maquinários agrícolas participam de todas as etapas da produção, desde a preparação do solo, plantio até a colheita e transporte, automatizando e agilizando processos, se tornando itens essenciais para a produção, cada maquinário possuindo suas particularidades e tecnologias, permitindo ao agricultor escolher um específico para atender sua demanda.

O emprego de maquinário agrícola, cada vez mais crescente, proporciona ao agricultor eficiência e produtividade no uso da terra. Tendo como resultado, vantagens competitivas na comercialização da produção. Para que isso se mantenha, é necessário garantir o bom funcionamento das máquinas, logo, é preciso realizar manutenções preventivas em todo o maquinário. Segundo SPOHR (2008), a manutenção preventiva de máquinas agrícolas são procedimentos que têm como objetivo manter as máquinas em condições ideais para seu funcionamento e prolongamento da vida útil, através de revisões periódicas, lubrificação e ajustes, obtendo o controle do desgaste natural de algumas peças e protegendo os maquinários.

A manutenção preventiva é toda ação e medida para o controle e monitoramento, com o objetivo de reduzir ou impedir falhas no desempenho dos maquinários. Tais medidas, presentes no manual do proprietário, são orientações para a programação de manutenções a serem seguidas. Desta forma, prevenindo que o maquinário apresente falhas devido ao mau funcionamento de algum componente, mantendo-o em boas condições e assim, prolongando a vida útil (LAMBRECHT et al., 2015).

Conhecer as especificações técnicas de cada máquina e realizar as manutenções se torna uma tarefa difícil, considerando a grande variedade de máquinas existentes, além da possibilidade de haver diversas máquinas diferentes em uma única propriedade, exigindo dos proprietários um controle rigoroso.

Atualmente existem diversos *softwares* com o intuito de auxiliar seus usuários no controle de manutenções de seus veículos automotores, por exemplo o sistema Gestão de Frotas da desenvolvedora Produsoft (2013), onde como este a grande maioria dos *softwares* não foram desenvolvidos especificadamente para o setor rural, sendo denominados como *softwares* de gestão de frotas, possuindo varias funcionalidades e dificultando a usabilidade do sistema, gerando uma frustação por parte do usuário que não utiliza a maioria das funções disponibilizadas e prefere manter o controle das manutenções em planilhas ou papel.

Neste contexto, este trabalho propõe um sistema *web*, capaz de avisar o usuário sobre

manutenções preventivas, através de um controle de manutenções previamente cadastradas, em que o usuário deverá cadastrar o período de meses ou o número do hodômetro necessário para realizar tal manutenção, permitindo assim se obter um controle por um período de meses ou horas trabalhadas com o maquinário. Para obter o controle da manutenção por tempo, o sistema emitirá um alerta 30 dias antes e no dia em que a manutenção irá expirar. Para realização do controle através do hodômetro, o usuário deverá inserir periodicamente o número contido no hodômetro do maquinário no sistema, onde será subtraído o número atual pelo número do hodômetro de quando foi realizada a última manutenção, permitindo assim o sistema alertar se a manutenção está próxima a expirar ou já expirou. O usuário poderá consultar as manutenções pendentes ou que estão para vencer pesquisando o maquinário pelo ID ou descrição, sendo direcionado para uma tela de checagem das manutenções, podendo determinar se a manutenção foi realizada ou adiada. O sistema manterá um histórico de manutenções, disponibilizando relatórios sobre o desempenho e gastos de cada maquinário e também qual usuário realizou a manutenção. Inicialmente o sistema irá limitar-se somente a manutenção de maquinários dos tipos tratores e colheitadeiras, podendo no futuro ser expandido para os demais maquinários e implementos agrícolas.

Neste trabalho, tendo em vista que conceito de maquinário agrícola é toda máquina e equipamento utilizado na fabricação de um produto ou na execução de determinada atividade agrícola, o conceito da palavra maquinário será limitado a somente maquinários do tido tratores e colheitadeiras, que são os tipos de máquinas agrícolas que inicialmente o sistema irá trabalhar.

A ideia para o desenvolvimento do sistema proposto surgiu em uma conversa com um produtor rural da região, onde o mesmo ressaltou o quanto os maquinários de sua propriedade haviam apresentado defeitos momentos antes do período de safra, acarretando prejuízos financeiros e atrasos na produção. Identificou-se que o problema inicial estava relacionado a não realização das manutenções preventivas, devido à falta de controle que a propriedade geria sobre estas. Neste contexto, foi identificada a necessidade de implementar um sistema para o controle das manutenções.

O sistema tem como principal vantagem o gerenciamento de revisões de modo a evitar que o maquinário quebre ou apresente falhas, isso consequentemente traz os benefícios para o produtor como uma colheita mais rápida, aumento de produtividade, redução de custos, controle, gerenciamento e eliminação de informações preenchidas em papéis e planilhas, salvando o usuário de gastos significativos com manutenções corretivas e reduzindo surpresas na hora da utilização do maquinário.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Desenvolver um sistema para o controle de manutenções preventivas de colheitadeiras e tratores agrícolas.

1.1.2 Objetivos Específicos

• Analisar e especificar requisitos necessários ao desenvolvimento do sistema proposto.

• Desenvolver a prototipação das telas, para dar forma e estrutura ao programa proposto.

• Estruturar o banco de dados de acordo com o Diagrama Entidade-Relacionamento (DER).

• Realizar o gerenciamento da execução do desenvolvimento utilizando princípios da metodologia ágil Scrum.

• Desenvolver/codificar o sistema, de acordo com os requisitos levantados.

• Implementar testes unitários e de integração para garantir estabilidade do sistema.

• Validar a proposta de sistema, implementando-o e realizando experimentos a fim de certificar-se de que o sistema atende as necessidades e expectativas do usuário.

1.2 PLANEJAMENTO DO TRABALHO

O planejamento do que será desenvolvido pelo aluno, ao longo do período letivo, está descrito no cronograma da Quadro 1. Neste cronograma constam todas as atividades com seus respectivos prazos para o cumprimento.

Quadro 1 – Cronograma de Atividades.

Atividades	**Abr**	**Mai**	**Jun**	**Jul**	**Ago**
1. Defesa da proposta do TCC	X				
2. Revisão dos apontamentos da Banca	X				
3. Revisão bibliográfica	X	X			
4. Elaboração do projeto de TCC	X	X			
5. Defesa do projeto de TCC		X			
6. Desenvolvimento do sistema			X	X	X
7. Validação dos requisitos implementados			X	X	X
8. Teste do programa resultante					X
9. Escrita da Monografia de TCC			X	X	X
10. Elaboração da apresentação final					X
11. Defesa final do TCC					X

2 TRABALHOS CORRELATOS

Atualmente é possível encontrar sistemas com funções semelhantes ao do sistema proposto. Alguns destes sistemas possuem mínimas funções, não permitindo a inserção ou alteração de manutenções, outros sendo disponibilizados como um módulo de manutenção preventiva em conjunto com um sistema de gestão de frotas, possuindo muitas funcionalidades, como ordens de serviço, gerenciamento de estoque e controle de tráfego, deixando o sistema complexo e com funções que um agricultor não utilizaria. A seguir, serão descritos alguns desses sistemas.

2.1 SISTEMAS SIMILARES

2.1.1 Manutenção Tratores Agrícolas

O programa computacional para manutenção preventiva de tratores agrícolas foi publicado na revista científica internacional InterSciencePlace por Klaver, Garcia e Junior (2013), o software permite a execução de tarefas para determinação da manutenção preventiva de tratores agrícolas de forma simplificada usando a internet, sem a necessidade de instalação de programas específicos .

No ambiente, o usuário insere o número de horas do trator registrado em seu horímetro. Após a entrada dos dados, o programa disponibiliza a exibição do conjunto de manutenções que o operador deve realizar, de acordo com as recomendações do fabricante do trator, informando as operações e recomendações específicas.

O sistema limita-se a manutenções somente de tratores da marca Massey Ferguson, não permitindo a alteração ou inserção de manutenções, também não permite o cadastramento dos maquinários, consequentemente não disponibilizando um histórico de manutenções dos mesmos, funcionando basicamente como uma consulta às recomendações das manutenções de forma online.

Diferentemente do sistema proposto, em que os maquinários e as manutenções são cadastradas pelo usuário e os dados ficam armazenados, obtendo-se um histórico de cada máquina, possibilitando a geração de relatórios e auxiliando o usuário na tomada de decisões estratégicas sobre custos de cada máquina, manutenções recorrentes e usuários responsáveis pelos lançamentos.

2.1.2 Gestão de Frotas.

O sistema Produsoft de Gestão de Frotas é um sistema pago oferecido pela desenvolvedora Produsoft. O sistema possui vários módulos para o controle de frotas, ordens de serviço, gerenciamento de abastecimentos, módulo de manutenção, entre outros.

Segundo a desenvolvedora Produsoft (2013), através modulo de manutenção, o usuário pode realizar o cadastro de veículos, hodômetros, funcionários e fornecedores, e se contratado o modulo de almoxarifado, é possível o cadastramento de produtos e a geração de um plano de custos de manutenções.

Para facilitar o gerenciamento da manutenção, o sistema trabalha em cinco etapas. Na primeira etapa, o sistema divide os veículos e equipamentos conforme suas partes mais importantes, denominando cada uma destas partes de subsistemas. Na segunda etapa, em cada subsistema, é cadastrada uma lista de serviços referente ao mesmo, em que por exemplo, se tem uma lista de serviços referentes ao subsistema motor, outra lista de serviços para o subsistema de câmbio e assim por diante. O terceiro passo é definir quais serviços se aplicam a cada família de veículos, servindo como um filtro para diminuir o tamanho da lista de serviços disponíveis para o usuário pesquisar. Uma vez feito isso, serão cadastrados os planos de manutenções preventivas, conforme a Figura 1, em que cada plano é aplicado a uma família de veículos. Por fim, após definir os planos de manutenções o sistema estará apto a iniciar a abertura automática de ordens de Serviço.

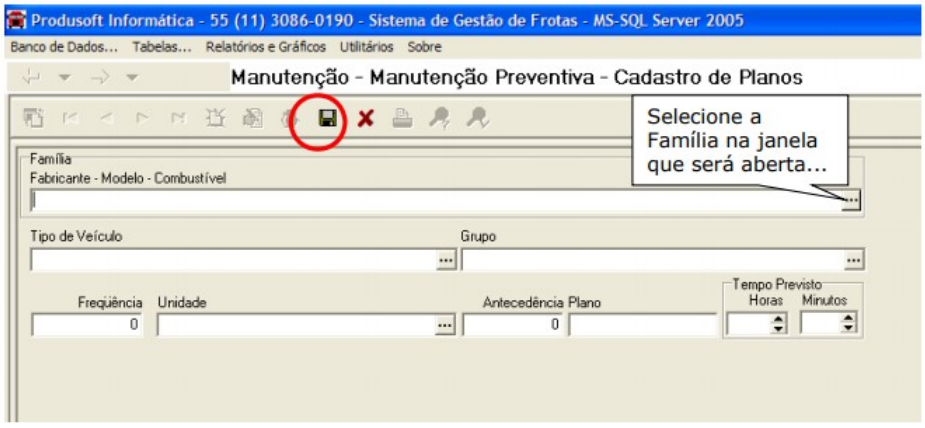


Figura 1 – Tela de cadastro de Planos - Gestão de Frotas.
Fonte: (PRODUSOFT, 2013)

Para a utilização do sistema, o usuário devera baixá-lo em seu computador, não sendo disponível o acesso ao sistema através de outros dispositivos como celulares ou *tablets*, sendo disponibilizado somente em conjunto com os demais módulos de gestão de frotas. O sistema torna-se complexo ao usuário que deseja realizar somente o controle das manutenções preventivas, pois terá que realizar um controle de ordens de serviço para que as manutenções funcionem de forma efetiva, uma vez que as manutenções só serão exibidas quando o usuário emitir uma ordem de serviço.

Em comparação ao sistema proposto, em que para se obter o controle das manutenções não é necessário o acoplamento com nenhum outro módulo ou sistema. Ademais, o sistema proposto será disponibilizado de forma *web*, ou seja, não será necessária a instalação do *software* sendo acessível a diferentes dispositivos que tenham acesso à internet.

2.1.3 Carro - Manutenção Programada

O aplicativo Carro Manutenção Programada foi desenvolvido pensando em facilitar o gerenciamento de veículos, garantindo que o usuário não perca nenhuma manutenção do seu carro e ainda possa acompanhar cada componente de seu carro individualmente (RM, 2021).

A aplicação determina manutenções programadas para troca de óleo, correia dentada, pastilhas de freio, troca dos pneus, alinhamento do veículo, troca do fluido de transmissão e troca das velas, podendo ser visualizado na Figura 2. Além disso, permite ao usuário o gerenciamento de vários veículos simultaneamente, de uma maneira fácil e prática, não exigindo do usuário nenhum conhecimento técnico sobre o assunto.

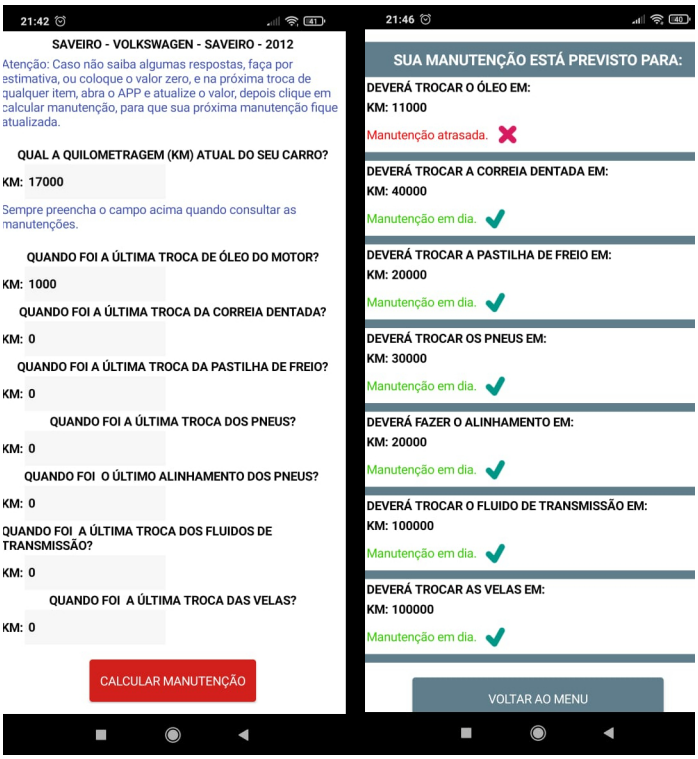


Figura 2 – Tela de resultados do aplicativo Carro
Fonte: RM (2021)

O aplicativo é disponibilizado somente para dispositivos Android e IOS, sendo permitido o cadastramento de vários veículos. O aplicativo mantém o histórico somente da última manutenção, para calcular quando deverá ser realizada a próxima, e não é permitida a alteração ou inserção de manutenções preventivas. Limitando-se ao uso de veículos utilitários e populares, sendo assim inviável utilizá-lo para o controle de manutenções preventivas de maquinários agrícolas, diferenciando-se do sistema proposto, em que o sistema será implementado especificadamente para maquinários agrícolas, e possibilitará a edição e inserção das manutenções pelo usuário.

2.2 DIFERENCIAL TECNOLÓGICO

O sistema proposto possui várias semelhanças entre os sistemas citados. Foi realizada uma comparação entre as principais funcionalidades dos sistemas descritos, ignorando funções de menor relevância para o sistema proposto, tendo como resultado a comparação apresentada no Quadro 2

Quadro 2 – Comparação de funções entre os sistemas.

Nome do sistema	**Pode ser utilizado para maquinários agrícolas**	**Edição e inserção de manutenções**	**Acessível em smartphones**	**Relatórios**	**Alertas**
Manutenção Tratores Agrícolas	X		X		
Produsoft - Gestão de Frotas	X	X		X	
Carro Manutenção Programada			X		
Software proposto	X	X	X	X	X

O sistema proposto possui características funcionais semelhantes aos demais sistemas, destacando-se pela proposta de alertas, para notificar o usuário de uma manutenção que foi previamente cadastrada, e também sendo implementados os pontos positivos destacados nos sistemas similares, reunindo as principais funcionalidades em um único sistema.

3 REFERENCIAL TEÓRICO

Neste capítulo serão descritas algumas tecnologias que podem ser aplicados no desenvolvimento de sistemas. Atualmente existem inúmeras ferramentas e linguagens de programação que visam auxiliar a produção de *softwares*, sendo que para cada problema abordado existem tecnologias que maximizam o desenvolvimento de um sistema, como por exemplo, a utilização de *frameworks*, que tornam o processo do desenvolvimento mais rápido.

No desenvolvimento de aplicações *web*, grande parte das aplicações são desenvolvidas baseadas em uma arquitetura monolítica, no qual toda a modularização utilizada é executada em uma mesma máquina e os módulos compartilham recursos de processamento, memória e arquivos. Assim as camadas de *frontend* e *backend* realizam uma comunicação entre si, onde toda parte de interface gráfica em que o usuário poderá interagir é contida no *frontend*, que envia requisições para o *backend*, responsável pelas regras de negócio e o tratamento dos dados, e onde são feitas as requisições para o banco de dados.

Para criação das interfaces gráficas existem diversas tecnologias que se destacam, como por exemplo, React (REACT, 2021), Vue (VUE, 2021), Angular (ANGULAR, 2021), JQuery (JQUERY, 2021) e Bootstrap (BOOTSTRAP, 2021), que visam auxiliar o desenvolvimento de páginas *web* forma simplificada e rápida. O Bootstrap é uma das tecnologias mais utilizadas, sendo um *framework* de código aberto, facilitando o desenvolvimento de HTML, CSS e JS. Assim auxiliando a criação de interfaces gráficas com tendências de design atraentes, que seguem padrões e princípios de usabilidade. O Bootstrap trabalha juntamente com o *framework* JQuery para ativar o comportamento de seus plug-ins Homehost (2021).

Para realizar a comunicação dos dados e a implementação das regras de negócio, existem diversas linguagens de programação que podem ser utilizadas, como por exemplo Python (PYTHON, 2021), Ruby (RUBY, 2021), Java (JAVA, 2021), PHP entre outros. Uma das linguagens mais utilizadas no mundo é o PHP. Segundo a revista Spectrum (2020), o PHP se encontra entre as 10 principais linguagens utilizadas para desenvolvimento *web*, conforme a Figura 3.

Criada por Rasmus Lerdorf em 1994, a linguagem de programação PHP (*Hypertext Preprocessor*) é uma linguagem de *script open source* de uso geral, sendo uma ferramenta que possibilita o pré-processamento de páginas HTML, conseguindo alterar o conteúdo de uma página antes de enviá-la para o navegador, formando assim páginas dinâmicas. Além disso, a linguagem também permite capturar entradas de dados do usuário, como formulários e outras formas de interação (BENTO, 2014). A linguagem PHP é uma das mais utilizadas para o desenvolvimento web, pela sua fácil integração com servidores, e possuir uma curva de aprendizado suave, se comparada a outras linguagens.



Language Ranking: IEEE Spectrum

Rank	Language	Type	Score
1	Python		100.0
2	Java		95.3
3	JavaScript		79.5
4	Go		73.1
5	Ruby		66.8
6	Dart		65.6
7	PHP		63.8
8	Scala		63.5
9	HTML		61.4
10	Kotlin		57.8

Figura 3 – Language Ranking 2020
Fonte: Spectrum (2020)

Independentemente da linguagem ou tecnologia, é de suma importância que o *software* seja desenvolvido no menor período de tempo e com a melhor qualidade possível. É neste contexto em que os *frameworks* se destacam, ajudando a agilizar o processo de desenvolvimento, de forma organizada e evitando repetições de código. O Laravel é um dos *frameworks* mais utilizados na atualidade, como pode ser observado na Figura 4. Muito se deve à sua simplicidade, sintaxe, flexibilidade e rica documentação.

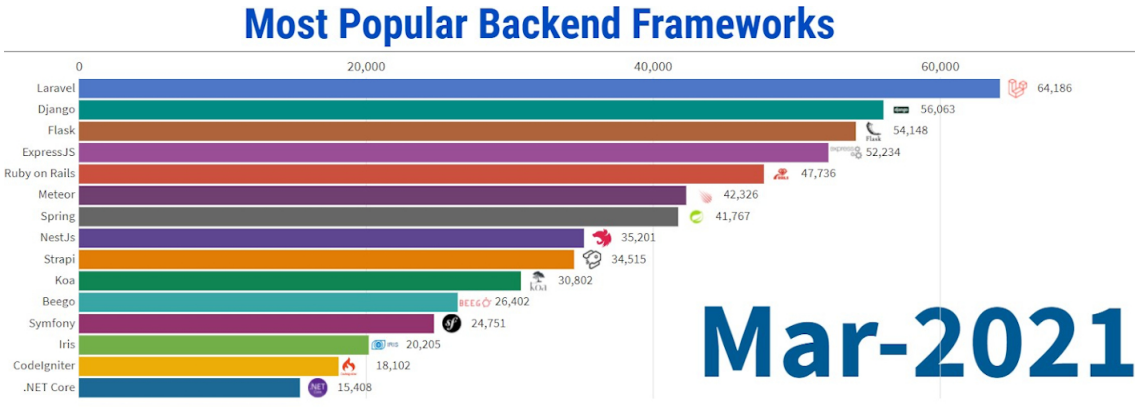


Figura 4 – Most Popular Backend Frameworks
Fonte: Statistics e Data (2021)

O framework utiliza arquitetura MVC, cujas responsabilidades, conforme a Figura 5, são dividias em 3 camadas: Model, View e Controller. A camada do Model é onde ficam todas as regras de negócio, sendo constituída pelas classes de acesso ao banco de dados. A camada View é a responsável por apresentar as páginas e resultados para o usuário, é a resposta que o *framework* envia para o navegador. A camada do Controller é responsável por cuidar e receber as requisições *web* e decidir o que fazer com elas. Nessa camada são definidos quais modelos devem ser executados, e para qual View será encaminhada a resposta. Esta camada é a que faz o link entre todas as outras (TURINI, 2015).

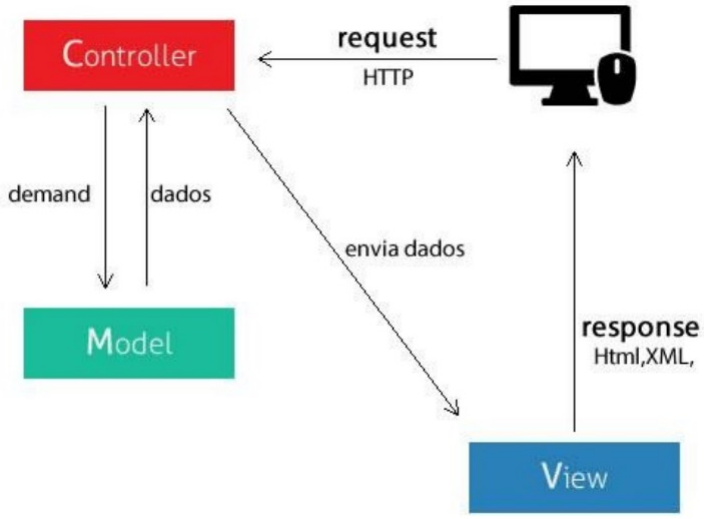


Figura 5 – Modelo de arquitetura MVC.
Fonte: Pelizza, Bertolini e Silveira (2018)

Para a criação de interface gráfica, o Laravel utiliza uma *engine* de *template* chamada Blade, que traz uma gama de ferramentas que ajudam a criar interfaces funcionais de forma rápida. Para resolver um grande problema de repetição, é criado um arquivo com o código comum, denominado de *layout* principal. Dessa forma, quando for necessário alterar ou adicionar qualquer item em partes em comum, edita-se um único lugar e todo o resto é atualizado, resultando em maior manutenibilidade.

Para que o sistema possa realizar a manipulação dos dados, é necessária a utilização de um banco de dados, que podem ser divididos em duas arquiteturas: relacionais e não relacionais. O banco de dados relacional possui uma forma intuitiva e direta de representar os dados em tabelas, armazenando e fornecendo acesso a pontos de dados que estejam relacionados entre si. Já o banco de dados não relacional é qualquer banco de dados que não siga o modelo relacional fornecido pelos sistemas tradicionais de gerenciamento de bancos de dados relacionais, em que os dados são representados como uma coleção de pares de valores-chaves (CARVALHO, 2015).

Existem diversos sistemas que podem ser utilizados para gerenciar bancos de dados, que são conhecidos como Sistemas de gerenciamento de bancos de dados (SGBD), como por

exemplo, Oracle Database, Microsoft SQL Server, IBM DB2, MariaDB, PostgreSql e um dos mais populares entre eles é o MySQL.

O MySQL é uma ferramenta gratuita e livre, que utiliza a linguagem Structured Query Language (SQL) para estabelecer a comunicação entre cliente e servidor. A linguagem SQL é a linguagem padrão utilizada pelos bancos de dados relacionais. O sucesso do MySQL deve-se à fácil integração com o PHP e com o seu comprovado desempenho, confiabilidade e facilidade de uso, tornando-se a primeira opção para aplicativos baseados na *web*, sendo utilizado pelos principais sites, entre eles o Spotify, Twitter, Netflix, Facebook e NASA (ORACLE, 2021).

E para o gerenciamento e execução do projeto, é crucial a escolha de uma metodologia para o processo de desenvolvimento, como o Scrum. Apesar da grande diversidade de metodologias presentes no mercado, a metodologia ágil Scrum é uma das mais utilizadas. Com uma proposta simples, vem sendo adotada por organizações de diversos tamanhos, desde multinacionais a startups. Segundo Audy (2015), O termo Scrum vem do jogo de rugby, é a jogada em que ficam todos juntos, cada um apoiando os demais frente a frente com time adversário, acontecendo sempre que a bola para ou sai de campo. Cada "Scrum" força com que os times se auto-organizem e reiniciem o jogo.

A metodologia Scrum é baseada em princípios semelhantes aos da Extreme Programming (XP), equipes pequenas, requisitos pouco estáveis ou desconhecidos e iterações curtas, para promover visibilidade para o desenvolvimento. O Scrum divide o desenvolvimento em iterações (*Sprints*) que variam de dois a quatro semanas de trabalho, em que cada equipe trabalha nos requisitos levantados no início da Sprint. Por meio de reuniões diárias e de curta duração (*daily scrum meeting*), são discutidas as dificuldades encontradas e identificados e resolvidos os fatores de impedimento. O ciclo de vida do método Scrum é dividido em dois momentos concorrentes: o de Discovery olhando mais à frente, definindo os próximos passos e a estratégia, e a fase de Delivery, para construção e entrega, conforme a Figura 6.



Figura 6 – Método Scrum
Fonte: Audy (2015)

A etapa inicial de **Discovery** é composta pelos passos de Visão, *Product Backlog*,

Release plan, Sprint backlog e início do *Sprint planning*. Nesta etapa é construída uma visão do produto desejado utilizando-se de técnicas colaborativas, por exemplo, reuniões de elicitação, critérios de aceitação e Mínimo Produto Viável (MVP), com o intuito de elicitar as necessidades reais do cliente e priorizá-las para constituição do *Release plan* e iniciar o *Product backlog*. A segunda etapa, **Delivery**, é a etapa desde o *Sprint planning* até a Retrospectiva, responsável pela construção com qualidade daquele conjunto de requisitos pactuados no *Sprint planning* e que definem o sucesso do *Sprint*.

3.1 METODOLOGIA DE DESENVOLVIMENTO

Nesta seção é descrito como será o fluxo de desenvolvimento do sistema proposto, considerando algumas metodologias e tecnologias já definidas para compor a etapa de desenvolvimento. A execução do projeto seguirá os seguintes passos:

- **Especificação dos requisitos.** Para a obtenção dos requisitos serão realizadas consultas com agricultores da região, com o objetivo de encontrar o maior número de características, funcionalidades e necessidades que o sistema deve conter. Cada *User Story* adquirida, representará apenas um requisito do *Product Backlog*.
- **Análise dos requisitos.** Após o levantamento dos requisitos, serão mapeados os objetivos e prioridades especificados. Esta fase, denominada de *Release Planning*, tem o objetivo de organizar as principais necessidades contidas no *Product Backlog*.
- **Prototipação e modelagem do sistema.** Com base nos requisitos levantados, serão feitas a prototipação das principais telas do sistema e a modelagem do banco de dados, podendo assim representar as especificações das informações.
- **Definições das Sprints.** Após o levantamento de todos os dados, cada Release será divida em uma ou mais *Sprints*, com o intuito de garantir frequência na entrega de no máximo a cada 2 a 4 semanas, de forma a manter um fluxo de melhoria contínua.
- **Desenvolvimento.** Com base os requisitos denominados como mais importantes e definidos no *Sprint backlog*, dar-se-á início ao desenvolvimento do sistema, utilizando as tecnologias citadas.
- **Review e Retrospectiva.** No final de cada *Sprint*, será realizado um *review* do que foi produzido com o objetivo de seguir alinhando os próximos passos. Na retrospectiva onde serão levantados os detalhes da *Sprint*, produtividade, problemas, erros e acertos, melhorando seu método e valores.
- **Deploy.** Essa etapa será executada ao final de cada *Sprint*, em que é disponibilizada a entrega daquilo que foi combinado, construído e concluído, onde o usuário final do sistema ira testar e dar a opinião sobre oque foi desenvolvido.

4 ANÁLISE E PROJETO DO SISTEMA

Neste capítulo serão apresentados os resultados preliminares da execução das primeiras etapas da metodologia de desenvolvimento, sendo a análise e especificação dos requisitos, a prototipação das principais telas do sistema e a modelagem do banco de dados.

4.1 LEVANTAMENTO DE REQUISITOS

O levantamento dos requisitos do programa foi realizado através de entrevistas com um gerente e um peão de uma fazenda da região. Foram realizadas visitas no local para melhor entendimento do ambiente onde o sistema será implementado e foram levantadas as User Storys através da utilização de "cartões", contendo três parâmetros: "QUEM", "O QUÊ" e "POR QUÊ". "QUEM" define quem é o usuário que tem a necessidade. "O QUÊ" define qual é a necessidade do usuário e "POR QUÊ" define qual o benefício do usuário ao ter a funcionalidade desenvolvida para atender a essa necessidade. O exemplo de um cartão é representado na Figura 7.

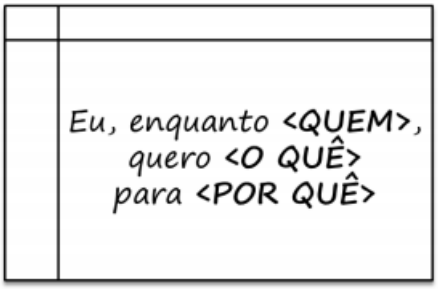


Figura 7 – Padrão de escrita de uma User Story.
Fonte: Audy (2015)

Cada user story representará um item do product backlog, representado no Quadro 3, que é uma lista contento tudo que será desenvolvido ao decorrer do projeto e ordenada de acordo com a importância para os clientes. A lista pode ser atualizada a cada momento e os itens do topo da lista são priorizados para que sejam desenvolvidos primeiro.

Quadro 3 – Product backlog.

Item do backlog	**Prioridade**
Criar Crud de usuários contendo os campos: nome, usuário, e-mail, senha e tipo do usuário (Administrador ou Agente).	1
Permitir que somente administradores realizem o cadastro de usuários do sistema.	2
Criar as tela de login com os campos usuário e senha.	3
Permitir que administradores e agentes possam efetuar o login no sistema.	4
Criar as telas de cadastramento de maquinários seguindo os mockups/protótipos em anexo.	5
Permitir que administradores cadastrem maquinários.	6
Permitir que agentes somente visualizem maquinários.	7
Criar a tela de seleção do maquinário com um campo de busca pelo ID ou descrição do maquinário, com um botão para redirecionamento a tela de cadastramento de manutenções e outro para tela de checagem de manutenções.	8
Criar as telas de cadastramento de manutenções pertencentes a um maquinário com os campos: descrição, período do hodômetro e o período de meses necessários para a realização da mesma, hodômetro e data da última vez que a manutenção foi realizada.	9
Criar as telas de lançamento do hodômetro seguindo os mockups/protótipos em anexo.	10
Criar a tela de checagem de manutenções seguindo os mockups/protótipos em anexo.	11
Criar a ação realizar manutenção com os campos preço e observação na tela de checagem de manutenções.	12
Permitir que os usuários possam realizar o adiantamento de manutenções futuras.	13
Criar a ação adiar manutenção com os campos: observação, período do hodômetro e período de meses que deseja postergar.	14
Criar tela inicial do sistema seguindo os mockups/protótipos em anexo.	15
Alertar os usuários na tela inicial do sistema sobre manutenções pendentes ou que estão para expirar.	16
Criar o relatório de manutenção exibindo a data de lançamento, descrição, observação, status, identificador e descrição do maquinário, usuário responsável pelo lançamento.	17
Criar tela de filtro do relatório de manutenções com os campos: usuário, maquinário, status da manutenção (realizada ou adiada), período em que foi lançada com a data inicial e data final.	18
Criar o relatório de gastos com manutenções exibindo a data de lançamento, descrição, observação, identificador e descrição do maquinário, usuário responsável pelo lançamento, preço da manutenção e o total geral das manutenções.	19
Criar tela de filtro do relatório de gastos com manutenções com os campos: usuário, maquinário, período em que foi lançada com a data inicial e data final.	20
Permitir que somente administradores possam filtrar os relatórios por usuário.	21
Ao cadastrar um maquinário, permitir a impressão de um QRCODE para facilitar o acesso a tela de lançamento do hodômetro.	22

4.2 PROTOTIPAÇÃO DAS TELAS

Esta etapa tem o objetivo de apresentar um protótipo contento o conceito de design e as telas do sistema, auxiliando no entendimento dos requisitos e apresentando conceitos e funcionalidades do sistema. Para o desenvolvimento do layout do programa, foi utilizado o template StartBootstrap (2020).

Na Figura 8, pode-se observar a tela inicial, contendo campos de usuário e senha necessários para assegurar um controle de acesso ao sistema, tanto o administrador quanto ao usuário comum efetuarão o login através da mesma tela.

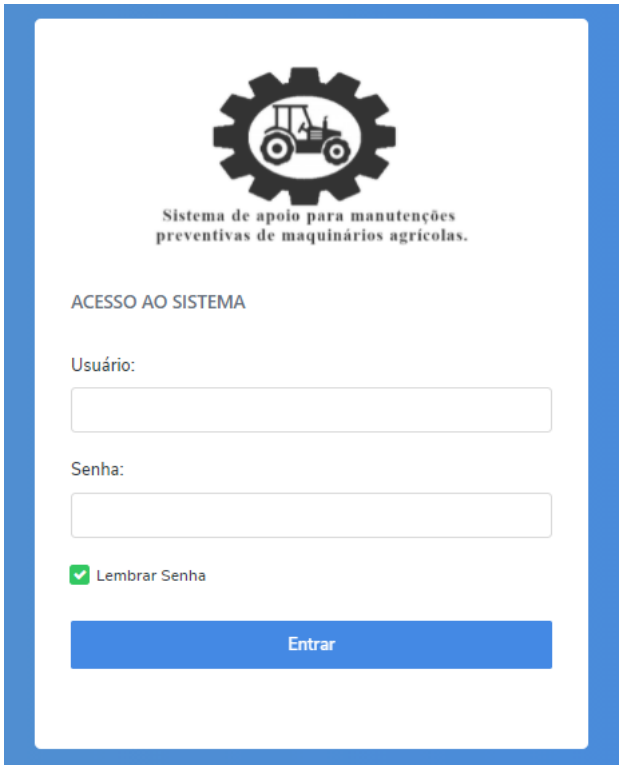


Figura 8 – Tela de Login.

Após efetuar o login, o usuário é redirecionado para tela inicial do sistema, representada pela Figura 9, exibe a quantidade de manutenções realizadas no mês atual e o total de gastos com estas manutenções, também contendo uma lista de avisos dos maquinários que estão com manutenções atrasadas e com manutenções que precisam ser realizadas em breve. Ao clicar em um item da lista o usuário será redirecionado para tela de checagem de manutenções, em que são exibidas as manutenções recorrentes daquele maquinário.



Figura 9 – Tela inicial do sistema.

Ao selecionar a opção usuários no menu lateral do sistema, o usuário será direcionado para tela de cadastro de usuários, conforme a Figura 10, está tela ficara disponível somente para administradores, para que ele possa cadastrar novos usuários que utilizarão o sistema, podendo selecionar o tipo de usuário para definir se será um usuário comum ou administrador.

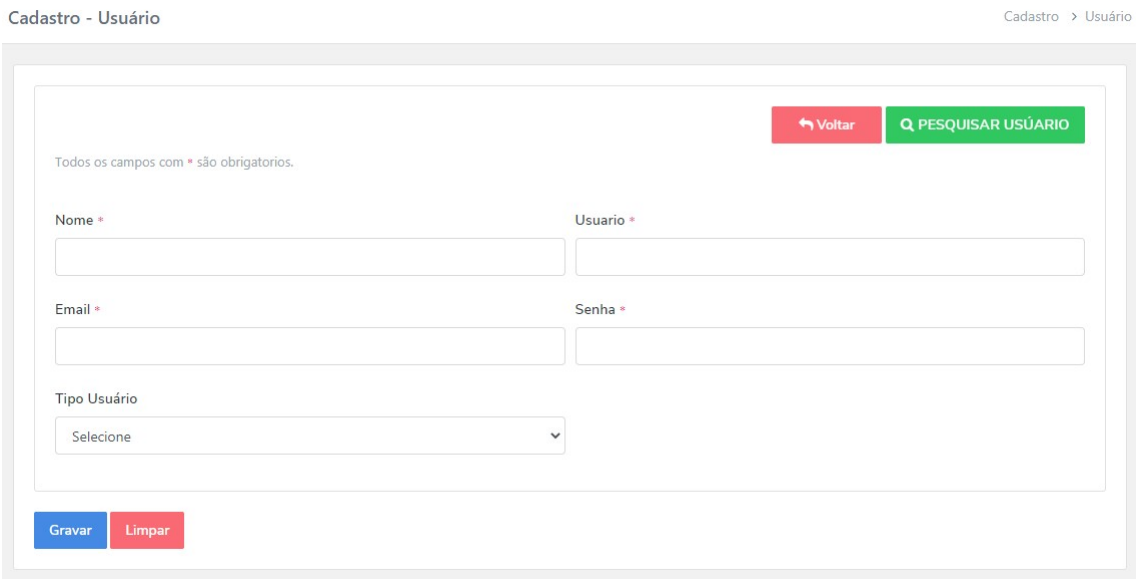


Figura 10 – Tela de cadastro de usuários.

Ao selecionar a opção maquinários no menu lateral do sistema, o usuário será direcionado para tela de cadastro de maquinários, conforme a Figura 11, tendo somente como preenchimento obrigatório os campos: descrição, sendo uma característica ou apelido dado ao maquinário, número do hodometrô que o maquinário se encontra no momento do cadastro e o número de identificação do maquinário. O número de identificação está colocado numa

plaqueta de alumínio, geralmente situada no chassi do lado esquerdo do maquinário, composto por uma sequência de letras e números únicos independente do fabricante ou modelo.

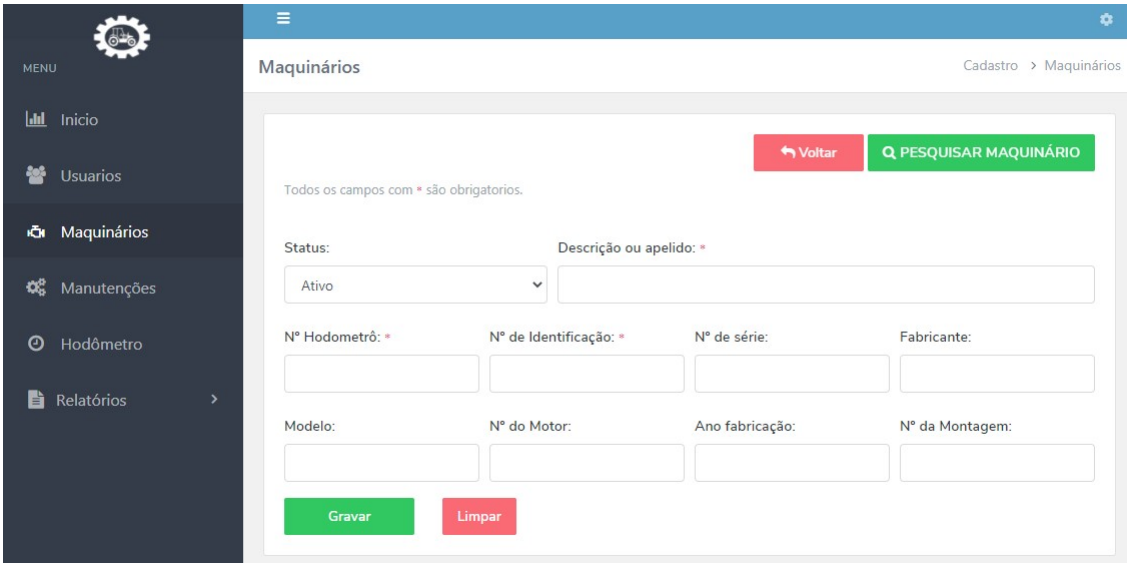


Figura 11 – Tela de cadastro de maquinários.

Após o cadastro o sistema disponibilizara a impressão de um QRCODE gerado a partir do número de identificação do maquinário, para facilitar a busca do mesmo no momento da realização do checklist das manutenções, conforme pode ser observado na Figura 12.



Figura 12 – Exemplo QRCODE para identificação do maquinário.

Ao realizar a leitura do QRCODE ou selecionar a opção lançar hodômetro no menu lateral do sistema, o usuário será direcionado para tela de lançamento do hodômetro, conforme a Figura 13. As manutenções são baseadas no número de horas do hodômetro ou um prazo médio estipulado no cadastro. Para que o sistema possa determinar se uma manutenção já deve ser realizada, o usuário deverá realizar a inserção periódica do número contido no hodômetro do

maquinário. Para realizar tal inserção, o usuário poderá selecionar o maquinário pesquisando-o pelo seu n° de identificação, descrição ou efetuar a leitura do QRCODE gerado pelo sistema. Após selecioná-lo, o sistema exibirá um modal com campo para inserção do n° do hodômetro, conforme a Figura 14. Após o lançamento, será exibido um alerta, como pode ser observado na Figura 15, informando ao usuário se o maquinário possui manutenções a serem feitas, podendo ser redirecionado para a tela de checagem de manutenções.



Figura 13 – Tela para pesquisa de um maquinário.



Figura 14 – Inserção do número do hodômetro.



Figura 15 – Salvamento hodômetro.

Para realizar o cadastramento ou a checagem das manutenções, o usuário deverá selecionar a opção manutenções no menu lateral do sistema, sendo redirecionado para a tela de seleção de maquinário, conforme a Figura 16, onde o usuário deverá pesquisar o maquinário em que deseja cadastrar ou checar as manutenções. Após selecionar a opção cadastrar manutenção na tabela de resultados do maquinário pesquisado, o usuário será redirecionado para tela de cadastramento de manutenções, conforme a Figura 17. A manutenção preventiva requer uma descrição do que será realizado, o período do hodômetro e quantidade de meses. Exemplo: troca de óleo necessária a cada 9000h ou 12 meses. Também são requeridos os dados da ultima realização da manutenção para o sistema conseguir determinar quando será necessária a realização da próxima. Estes dados serão atualizados automaticamente quando o usuário realizar a manutenção. Abaixo do formulário de cadastramento, o sistema exibe uma tabela de manutenções cadastradas do maquinário selecionado. Devido ao grande número de manutenções que um maquinário pode possuir, está tabela contem paginação e um campo de pesquisa para que o usuário possa editar e remover manutenções de forma rápida.



Figura 16 – Tela de seleção de maquinário.

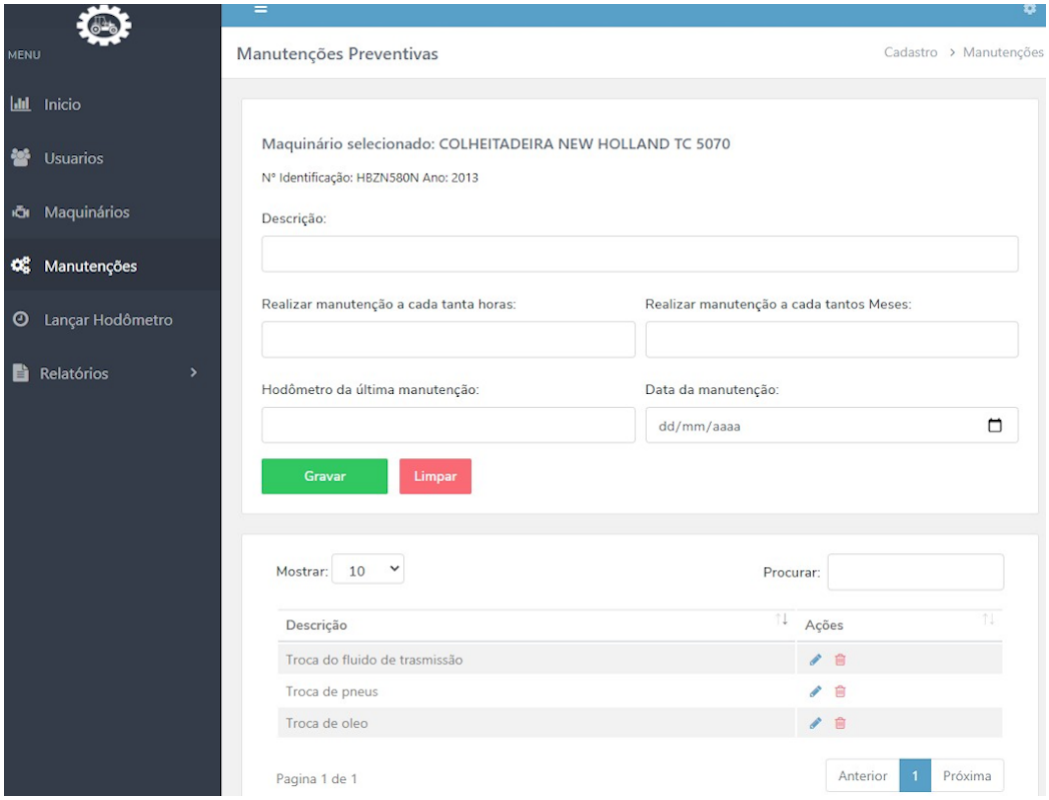


Figura 17 – Tela de cadastro de manutenções.

Após selecionar a opção checagem de manutenções na tabela de resultados do maquinário pesquisado ou clicar em um item da lista de avisos da tela inicial do sistema, o usuário será redirecionado para tela de checagem de manutenções, conforme a Figura 18. Nesta tela são exibidas as manutenções preventivas necessárias do maquinário selecionado. A exibição é dada se o maquinário atingiu o limite de horas ou prazo de meses estipulado no cadastro da manutenção. A coluna da tabela ficará em vermelho caso a manutenção já tenha ultrapassado o limite do hodômetro ou o prazo de meses, já em casos em que a manutenção estiver prestes a atingir os limites à coluna ficará marcada na cor amarela. Permitindo assim ao usuário a identificação das manutenções que já expiraram ou irão expirar. Também é exibida uma tabela com uma lista geral de manutenções do maquinário selecionado, caso o usuário deseje antecipar a realização. Através destas tabelas o usuário poderá selecionar se efetuou a realização ou adiamento da manutenção.

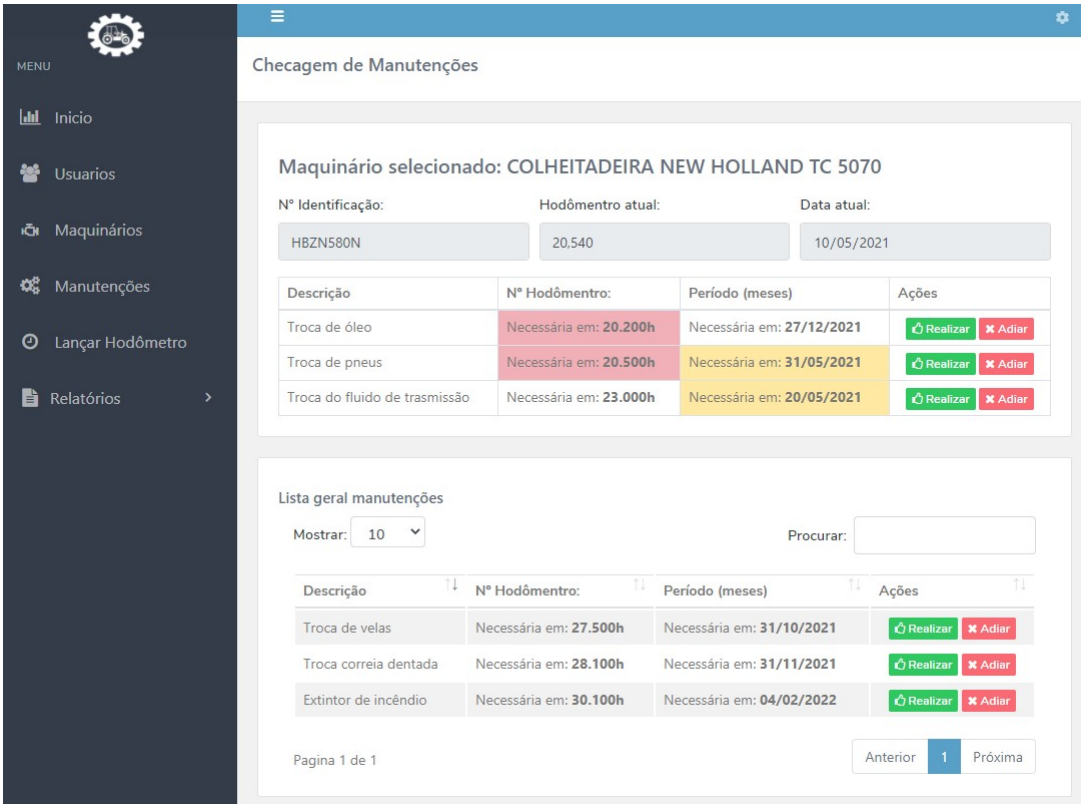


Figura 18 – Tela de checagem de manutenções.

Ao efetuar a realização da manutenção o usuário poderá preencher uma observação e o preço, campos necessários para se obter um controle dos gastos com cada maquinário, como pode ser visto na Figura 19. Ao efetuar o adiamento da manutenção, representado na Figura 20, o usuário deverá determinar a quantidade de horas e meses que deseja adiar a realização da manutenção, juntamente com uma observação explicando o motivo do adiamento.

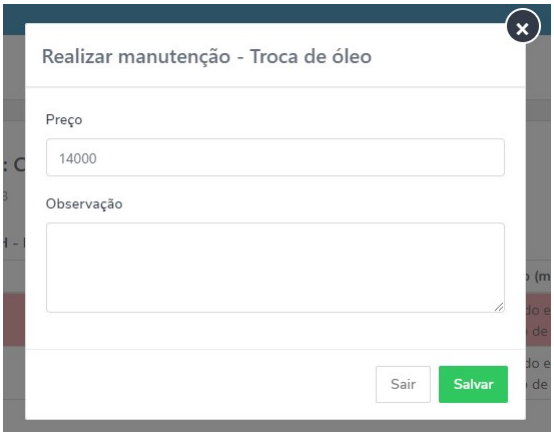


Figura 19 – Modal com formulário da realização da manutenção.

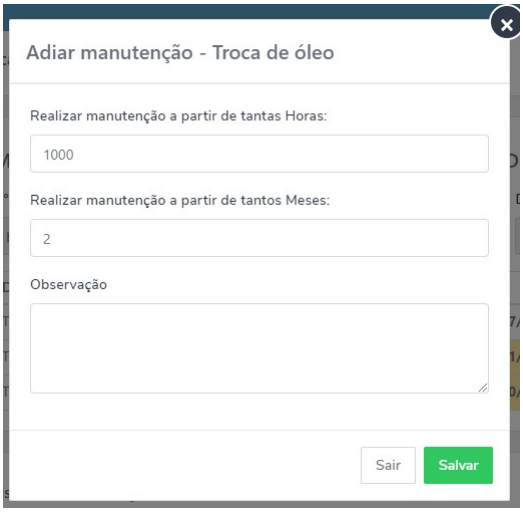


Figura 20 – Modal com formulário de adiamento da manutenção.

Ao selecionar a opção relatórios no menu lateral do sistema, será exibido um submenu com as opções: relatório de gastos com manutenções e relatório de manutenções. Após selecionar a opção de relatório de manutenções, o usuário será redirecionado para tela de filtros do relatório de manutenções, conforme a Figura 21. Nesta tela o usuário poderá filtrar pelo usuário responsável pela manutenção, pelo status da manutenção se ela foi adiada ou realizada, por um maquinário específico e também por um período de lançamento informando a data inicial e final. O relatório gerado será aberto em uma nova guia do navegador em formato PDF, contendo os dados: data de lançamento, descrição, observação e status da manutenção e o usuário responsável pelo lançamento, também o número de identificação do maquinário e descrição, conforme a Figura 22.

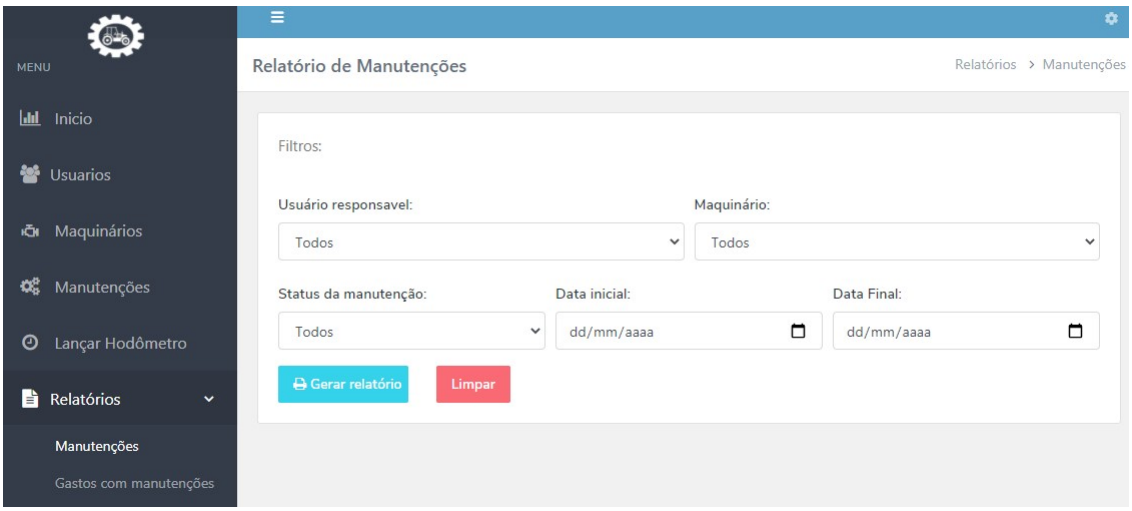


Figura 21 – Tela de geração do relatório de manutenções.

SISTEMA DE APOIO PARA MANUTENÇÕES PREVENTIVAS DE MAQUINÁRIOS AGRÍCOLAS
Relatório geral de manutenções

Filtrando entre 01/01/2021 e 30/06/2021

Data	Descrição	Observação	Status	ID	Maquinário	Usuário
11/02/2021	Troca de óleo	Utilizado óleo EVORA TURBO MA – API CI-4	Realizada	HBZN580N	COLHEITADEIRA JOHN DEERE S790	Lucas Samuel
15/03/2021	Troca do fluido de transmissão	Fluido Shell Spirax S4 TXM	Realizada	HBZN580N	COLHEITADEIRA JOHN DEERE S790	Lucas Samuel
19/04/2021	Troca de Pneus	Os pneus ainda estavam em bom estado	Adiada	HQEQWE5	COLHEITADEIRA MASSEY FERGUSON 5650	Lucas Samuel
29/05/2021	Troca Correia V	Modelo correia 301146A1 CNH	Realizada	QEQWE58	Colheitadeira New Holland	João Maria
25/06/2021	Troca de Fluido de Freio	Fluido de Freio RBF 660 Motul	Realizada	345SDDR23	COLHEITADEIRA CASE 4130	Lucas Samuel

Figura 22 – Exemplo do relatório de manutenções.

Ao selecionar a opção gastos com manutenções no submenu de relatórios, o usuário será redirecionado para tela de filtros do relatório de gastos com manutenções, conforme a Figura 23. Nesta tela o usuário poderá filtrar pelo usuário responsável pelo lançamento da manutenção, por um maquinário específico ou pelo período de lançamento informando uma data inicial e final. O relatório gerado será aberto em uma nova guia do navegador em formato PDF, conforme a Figura 24. Contendo os dados: o número de identificação e descrição do maquinário, a descrição da manutenção, preço, observação, data de realização e o usuário responsável pela manutenção. Contento no final do relatório um total geral das manutenções que foram filtradas.

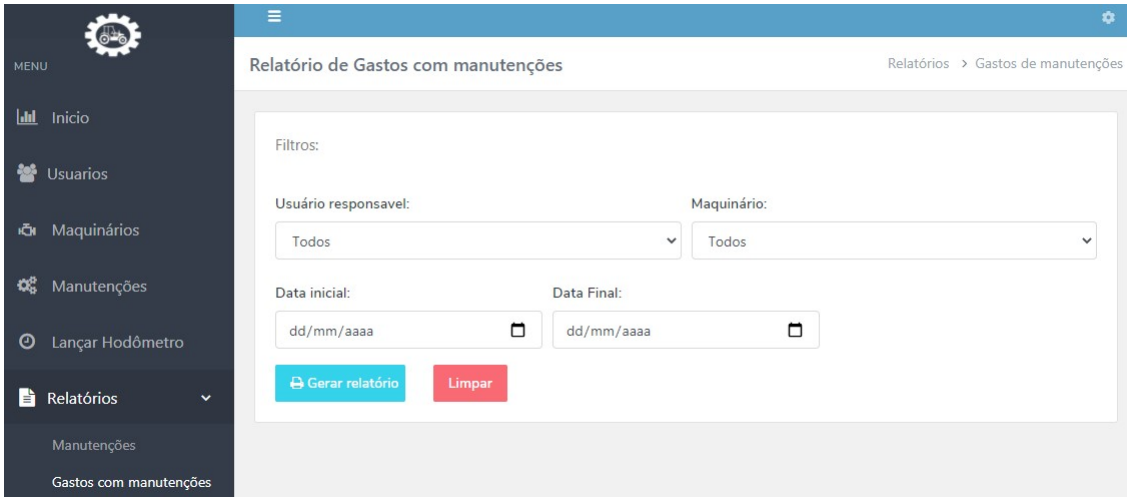


Figura 23 – Tela de geração do relatório de gastos com manutenções.



SISTEMA DE APOIO PARA MANUTENÇÕES PREVENTIVAS DE MAQUINÁRIOS AGRÍCOLAS
Relatório de Gastos com manutenções

Filtrando entre 01/01/2021 e 30/06/2021

Data	Descrição	Observação	ID	Maquinário	Usuário	Preço R$
11/02/2021	Troca de óleo	Utilizado óleo EVORA TURBO MA – API CI-4	HBZN580N	COLHEITADEIRA JOHN DEERE S790	Lucas Samuel	575,00
15/03/2021	Troca do fluido de transmissão	Fluido Shell Spirax S4 TXM	HBZN580N	COLHEITADEIRA JOHN DEERE S790	Lucas Samuel	485,00
29/05/2021	Troca Correia V	Modelo correia 301146A1 CNH	QEQWE58	Colheitadeira New Holland	João Maria	80,00
25/06/2021	Troca de Fluido de Freio	Fluido de Freio RBF 660 Motul	345SDDR23	COLHEITADEIRA CASE 4130	Lucas Samuel	259,00
					Total Geral:	**1.399,00**

Figura 24 – Exemplo do relatório de gastos com manutenções.

4.3 MODELAGEM DO BANCO

O banco foi desenhado para poder atender todos os requisitos levantados e tornar mais fácil a execução de alterações para melhorias futuras. A estrutura do banco de dados foi representada por um Diagrama Entidade-Relacionamento (DER), que consiste em um tipo de fluxograma que ilustra como as entidades se relacionaram entre si dentro do sistema, conforme pode ser observado na Figura 25.

O banco de dados escolhido para o sistema foi o MySQL, devido sua estabilidade, confiabilidade, alta disponibilidade para uso em sistemas *web* e ser um Banco de Dados relacional de código aberto, possuindo uma grande comunidade de usuários e administradores.

Dentre as tabelas, é conveniente esclarecer a funções e os relacionamentos das mais importantes para o sistema:

- **Machines:** Tabela em que serão armazenados os dados dos maquinários, possuindo como atributos mais importantes: id (chave primária), identificationNumber (número de identificação do maquinário) , description (descrição do maquinário), hodometro (número do hôdometro do maquinário).
- **Maintenance:** Tabela em que serão armazenadas as manutenções. Possuindo os atributos: id (chave primária), description (descrição da manutenção), range-hodometro (quantidade de horas necessárias para realizá-la), range-months (período de meses necessários para realizá-la), last-hodometro (número do hôdometro da última vez em que ela foi realizada), last-months (data de sua última realização), user-id (usuário responsável pelo cadastro, chave estrangeira com relacionamento N para1 (Muitos para Um) onde cada manutenção pode pertencer a um único usuário, mas um usuário pode possuir várias manutenções.
- **Maintenance-check:** Tabela em que serão armazenadas as manutenções realizadas. Possuindo os atributos: id (chave primária), price (preço), note (observação), number-hodometro (número do hôdometro de quando foi realizada), created-at (data de lança-

mento), maintenance-id (id da manutenção, chave estrangeira com relacionamento N para 1 (Muitos para Um) onde cada manutenção pode pertencer a várias checagens, mas uma checagem deve ser somente de uma manutenção), user-responsible (usuário responsável pelo lançamento, chave estrangeira com relacionamento N para 1 (Muitos para Um) onde cada checagem pode pertencer a um usuário, mas um usuário pode possuir várias checagens.

- **Maintenance-postpone:** Tabela em que serão armazenadas as manutenções adiadas. Possuindo os atributos: id (chave primária), note (observação), postpone-months (período de meses que será postergado), postpone-hodometro (numero hôdometro que será postergado), maintence-id (id da manutenção, chave estrangeira com relacionamento N para 1 (Muitos para Um) onde cada manutenção pode pertencer a vários adiamentos, mas um adiamento deve ser somente de uma manutenção), user-responsible (usuário responsável pelo lançamento, chave estrangeira com relacionamento N para 1 (Muitos para Um) onde cada adiamento pode pertencer a um usuário, mas um usuário pode possuir vários adiamentos.
- **Users:** Tabela em que serão armazenados os usuários do sistema. Possuindo os atributos mais importantes: id (chave primária), name (nome do usuário), password (senha), type-user (tipo do usuário administrador ou agente).

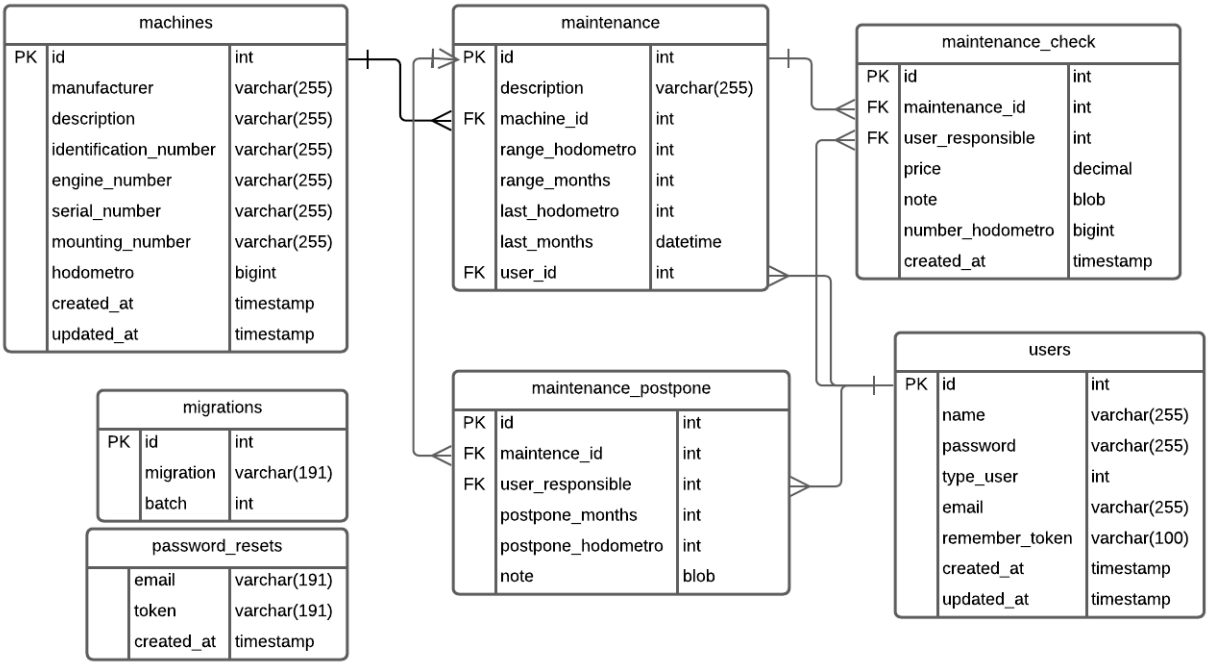


Figura 25 – Diagrama de Entidade Relacionamento.

5 CONCLUSÃO

Este trabalho apresentou os desafios e objetivos no desenvolvimento de um Sistema de Apoio para Manutenções Preventivas de Maquinários Agrícolas, que poderá ser utilizado por usuários que não possuem um vasto conhecimento tecnológico. A partir do sistema proposto, torna-se possível o gerenciamento de manutenções preventivas de forma simplificada, utilizando a internet, sem necessidade de instalação de programas específicos e podendo ser acessado tanto por meio de um computador quanto de um smartphone.

O sistema deverá ser de fácil usabilidade, capaz de instruir e auxiliar o agricultor sobre o controle de manutenções periódicas, em que o usuário irá previamente cadastrar as manutenções, definindo o período de checagem de meses ou de acordo com o número do hodômetro, possibilitando ao sistema alertar sobre a necessidade de efetuar determinada manutenção, cabendo ao usuário definir a realização ou adiamento da mesma, mantendo um histórico de manutenções e disponibilizando relatórios sobre o desempenho e gastos de cada maquinário.

O sucesso da ferramenta é importantíssimo para a redução da apresentação de falhas devido ao mau funcionamento de algum componente do maquinário, mantendo-os em boas condições, com as revisões periódicas em dia e assim prolongando sua vida útil.

Referências

ANGULAR. **Introduction to the Angular Docs**. 2021. Disponível em: <https://blog.angular.io/introducing-angular-devtools-2d59ff4cf62f>. Acesso em: 22 de maio de 2021. Citado na página 8.

AUDY, J. **Scrum 360: Um guia completo e prático de agilidade**. São Paulo: Editora Casa do Código, 2015. Citado 2 vezes nas páginas 11 e 13.

BENTO, E. J. **Desenvolvimento web com PHP e MySQL**. São Paulo: Editora Casa do Código, 2014. Citado na página 8.

BOOTSTRAP. **Build fast, responsive sites with Bootstrap**. 2021. Disponível em: <https://getbootstrap.com/>. Acesso em: 22 de maio de 2021. Citado na página 8.

CARVALHO, V. **MySQL: Comece com o principal banco de dados open source do mercado**. São Paulo: Editora Casa do Código, 2015. Citado na página 10.

HOMEHOST. **O que é Bootstrap: Tudo sobre este Framework**. 2021. Disponível em: <https://www.homehost.com.br/blog/tutoriais/o-que-e-bootstrap/>. Acesso em: 11 de maio de 2021. Citado na página 8.

JAVA. **O que é o Java?** 2021. Disponível em: <https://www.java.com/pt-BR/about/whatis_java.jsp>. Acesso em: 22 de maio de 2021. Citado na página 8.

JQUERY. **jQuery Documentation**. 2021. Disponível em: <https://jquery.com/>. Acesso em: 22 de maio de 2021. Citado na página 8.

KLAVER, P. P. C.; GARCIA, R. F.; JUNIOR, J. F. S. V. Programa computacional para determinar a manutenção preventiva de tratores agrícolas. **Revista Científica Internacional**, v. 1, n. 7, 2013. ISSN 1679-9844. Disponível em: <http://www.interscienceplace.org/isp/index.php/isp/article/view/242>. Citado na página 4.

LAMBRECHT, E. et al. Nota técnica: custos de manutenção preventiva para tratores em função do tipo de tração e categoria do sistema hidráulico - doi: 10.13083/1414-3984/reveng.v23n3p270-277. **Revista Engenharia na Agricultura - Reveng**, v. 23, n. 3, p. 270–277, jul. 2015. Disponível em: <https://periodicos.ufv.br/reveng/article/view/476>. Citado na página 1.

ORACLE, C. **MySQL Customers**. 2021. Disponível em: <https://www.mysql.com/customers/>. Acesso em: 09 de maio de 2021. Citado na página 11.

PELIZZA, A. C.; BERTOLINI, C.; SILVEIRA, S. R. Um estudo sobre técnicas de teste de software no framework laravel. **RESIGeT - Revista Eletrônica de Sistemas de Informação e Gestão tecnológica**, v. 9, n. 3, 2018. ISSN 2237-0072. Disponível em: <http://periodicos.unifacef.com.br/index.php/resiget/article/view/1593/1168>. Citado na página 10.

PRODUSOFT. **Sistema Produsoft de Gestão de Frotas.** 2013. Disponível em: <https://www.produsoft.com.br/gestao-frota-agricola>. Acesso em: 18 de março de 2021. Citado 2 vezes nas páginas 1 e 5.

PYTHON. **Python documentation**. 2021. Disponível em: <https://www.python.org/>. Acesso em: 22 de maio de 2021. Citado na página 8.

REACT. **NA JavaScript library for building user interfaces**. 2021. Disponível em: <https://reactjs.org/>. Acesso em: 22 de maio de 2021. Citado na página 8.

RM, c. **Carro – Manutenção Programada**. 2021. Disponível em: <https://play.google.com/store/apps/details?id=rm.manutencao.criacoes>. Acesso em: 18 de março de 2021. Citado na página 6.

RUBY. **Sobre o Ruby**. 2021. Disponível em: <https://www.ruby-lang.org/pt/about/>. Acesso em: 22 de maio de 2021. Citado na página 8.

SPECTRUM, I. **Interactive: The Top Programming Languages**. 2020. Disponível em: <https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2020>. Acesso em: 23 de março de 2021. Citado 2 vezes nas páginas 8 e 9.

SPOHR, R. B. **Manutenção de Máquinas Agrícolas.** 2. ed. Itaqui: Universidade Federal do Pampa – Unipampa, 2008. Citado na página 1.

STARTBOOTSTRAP. **SB Admin 2**. 2020. Disponível em: <https://startbootstrap.com/theme/sb-admin-2>. Acesso em: 22 de maio de 2021. Citado na página 15.

STATISTICS; DATA. **Most Popular Backend Frameworks – 2012/2021**. 2021. Disponível em: <https://statisticsanddata.org/data/most-popular-backend-frameworks-2012-2021/#pll_switcher>. Acesso em: 09 de maio de 2021. Citado na página 9.

TURINI, R. **PHP e Laravel: Crie aplicações web como um verdadeiro artesão**. São Paulo: Editora Casa do Código, 2015. Citado na página 10.

VIAN, L. G. B. e C. E. A relação entre a demanda de máquinas agrícolas e a política agrícola. **Revista de Política Agrícola**, v. 26, n. 1, 2017. ISSN 2317-224X. Disponível em: <https://seer.sede.embrapa.br/index.php/RPA/article/view/1244>. Citado na página 1.

VUE. **The Progressive JavaScript Framework**. 2021. Disponível em: <https://vuejs.org/>. Acesso em: 22 de maio de 2021. Citado na página 8.