

Aplicação web para interfaceamento de métodos de retificação

Web application for rectification methods interfacing

RESUMO

Laércio Regiane Júnior
laercioj@alunos.utfpr.edu.br
Universidade Tecnológica Federal
do Paraná, Medianeira, Paraná,
Brasil

Nelson Miguel Betzek
nmbetzek@utfpr.edu.br
Universidade Tecnológica Federal
do Paraná, Medianeira, Paraná,
Brasil

A agricultura de precisão (AP) consiste em um conjunto de técnicas modernas com o objetivo de otimizar o uso de insumos. Para isto é preciso dividir a área de produção em Zonas de Manejo (ZM), cada uma possuindo propriedades do solo homogêneas. Muitas vezes, entretanto, estas ZMs podem ter contornos irregulares dificultando a aplicação da AP. Assim, métodos de retificação foram concebidos em estudos científicos, e o objetivo deste trabalho é criar uma aplicação que sirva de interface para estes métodos, focando principalmente na usabilidade. Para isso, foi utilizado o *framework angular*, uma plataforma de desenvolvimento web com suporte a *single page applications*, bem como a biblioteca angular material, fornecendo componentes de implementação facilitada para a interface gráfica do projeto. Ao final do período de programação obteve-se uma aplicação com interfaces intuitivas ao usuário e que permitem utilizar a retificação sem a necessidade de conhecimentos prévios de computação.

PALAVRAS-CHAVE: *AngularJS. Software. Zona de Manejo.*

ABSTRACT

Recebido: 19 ago. 2020.

Aprovado: 01 out. 2020.

Direito autoral: Este trabalho está licenciado sob os termos da Licença Creative Commons-Atribuição 4.0 Internacional.



The precision agriculture (PA) is a set of modern techniques with the objective of optimizing the use of agricultural inputs. For that it is needed to divide the production area in management zones (MZ), each one having homogeneous soil properties. Oftentimes, however, these MZs may have irregular outlines making it hard to apply the PA. Therefore, rectification methods were designed in scientific studies, and this work's objective is to create an application to be used as an interface for those methods, focusing mainly on usability. For this, the angular framework was used, a web development platform with support for single page applications, as well as the angular material library, providing easy for implementation components to the project's graphical interface. At the end of the programming cycle it was obtained an application with intuitive user interfaces and that allow the rectification to be used with no need of previous computation knowledge.

KEYWORDS: *AngularJS. Software. Management Zones.*



INTRODUÇÃO

A agricultura é um ramo de atividades muito antigo que busca produzir recursos para sustentar a população mundial através do plantio, gerando matéria prima para diversas indústrias, principalmente a alimentícia (HARRIS; FULLER, 2014, p. 104 - 113). A variabilidade dos nutrientes do solo em uma única área de plantio é um problema recorrente, geralmente devido a fatores como clima, processos geológicos, topografia e quantidade de matéria orgânica. Esta variabilidade reduz a eficácia da aplicação de recursos, pois, diferentes talhões podem necessitar de quantidades diferentes de adubo ou minerais (BETZEK et al., 2018).

A fim de contornar este desafio é possível fazer uso da Agricultura de Precisão (AP), que é um sistema de gestão da agricultura baseado nas variações espaciais das características do solo, assim, a aplicação de insumos é feita de forma otimizada e localizada, visando aumentar a produtividade e diminuir a quantidade de recursos gastos (SOBJAK et al., 2013). A AP se baseia na divisão do campo de cultivo em sub áreas chamadas de Zonas de Manejo (ZM).

Cada ZM é homogênea em indicativos como quantidade de colheita e características químicas e físicas do solo, permitindo assim a aplicação de recursos de forma personalizada e específica (SOBJAK et al., 2013). Porém, é necessário aperfeiçoar as técnicas de delineamento das ZMs a fim de melhorar a divisão da área, bem como reduzir a fragmentação. Com este objetivo foram implementados métodos computacionais de retificação e suavização, que geram mapas com áreas melhor delimitadas, proporcionando aplicação otimizada da AP (BETZEK et al., 2018). Estas rotinas apresentaram resultados positivos, porém foram concebidas com fim acadêmico. Assim, este projeto de pesquisa tem como objetivo desenvolver uma ferramenta computacional para tornar os métodos de retificação mais acessíveis para usuários de outros âmbitos, como: pesquisadores, alunos e profissionais (gestor, técnico, engenheiro, consultor, empresários,) com foco na área agrícola.

Alguns quesitos tem maior atenção como é o caso da usabilidade e a experiência dos usuários. Desta maneira, foi implementada uma aplicação web para servir de interface das rotinas de delineamento implementadas anteriormente (BETZEK et al., 2018). Não há necessidade de cadastros ou login, isto aliado com a disponibilização na internet simplifica e facilita a utilização da ferramenta computacional proposta. Os resultados obtidos por meio do aplicativo desenvolvido foram relevantes. Ao final da execução de todos os procedimentos propostos, o usuário tem a possibilidade de visualizar mapas gerados através da especificação do método de retificação e também das zonas de manejo utilizadas.

MATERIAL E MÉTODOS

Para o desenvolvimento da aplicação foram utilizados os conceitos de *Single Page Applications (SPAs)*, na qual o processamento de atualização das páginas web é feito em *client-side*, os servidores são apenas responsáveis por fornecer dados e não interfaces. Assim, as páginas web também sofreram reformulações, e, agora são formadas por componentes independentes que serão atualizados de maneira

individual, ou seja, não existe necessidade de recarregar a *view* por completo, apenas as partes alteradas pelo usuário (OLIVEIRA, 2017, p. 6-7).

Para obter acesso a estas funcionalidades a implementação do projeto foi feita no *framework angular*, que integra o conceito de *SPAs* a partir da junção de códigos em *HTML (Hypertext Markup Language)*, *CSS (Cascading Style Sheet)* e *TypeScript*. Assim, os dois primeiros são responsáveis pela interface de usuário da aplicação, enquanto o último fornece a lógica de funcionamento.

O *HTML* é uma linguagem de marcação de texto utilizada na *World Wide Web* desde a criação desta em 1990. Sua maior funcionalidade para a época era permitir a publicação global de arquivos, sendo estruturada como uma linguagem que a maioria dos computadores poderiam processar. Ela permite a organização de texto na forma de tabelas, cabeçalhos e listas, bem como pode empregar vídeos, fotos e planilhas como parte de sua composição. Toda sua organização é feita por meio de *tags*, determinando tanto a abrangência das estruturas citadas, bem como o seu conteúdo.

O *CSS* é uma página de código responsável por personalizar a aparência de uma página *web*, atrelando estas alterações às *tags HTML*. Ele passou a se tornar padrão no desenvolvimento de aplicações para a *internet* a partir de 1996, possuindo mais de 50 atributos modificáveis, como tamanho, cor e tipo da fonte do texto, tamanho das bordas de um elemento, texto em itálico ou negrito e o alinhamento do conteúdo. Para facilitar sua utilização ainda é possível a criação de classes com atributos preestabelecido e que poderão ser reutilizadas se necessário, permitindo reaproveitamento de código (MEYER, 2004, p. 3-5).

O *TypeScript* é uma linguagem de programação multiparadigma criada pela *Microsoft*, que funciona como um superconjunto do *Java Script*, ou seja, ela implementa todas as funcionalidades desta, porém, adiciona um sistema de tipagem estática e alguns outros recursos.

Junto do *framework* foi também empregada a biblioteca *angular material*, criada e utilizada pela *Google*, que disponibiliza componentes para a interação com o usuário, como campos de formulário e botões customizáveis, e para a apresentação de dados, como tabelas e telas de *pop-up*. Sua implementação é simples e a criação dos componentes se dá por meio de *tags*, assim como no *HTML*, porém, sua lógica de funcionamento fica por conta da programação no *TypeScript*. Além disto também é possível personalizá-los através do *CSS*.

Para exibir e gerar os mapas temáticos foi utilizada a biblioteca *OpenLayers*, que é implementada em *Java Script*, de código aberto e desenvolvida com auxílio da comunidade, permitindo ao programador manipular e trabalhar com mapeamento por meio de *back-ends* como os oferecidos pelo *Google maps* e *WMS*, por exemplo (HAZZARD, 2011).

Todos os testes e o desenvolvimento foram feitos focando-se no navegador *Google Chrome*.

RESULTADOS E DISCUSSÃO

O *front-end* da aplicação foi desenvolvido como uma interface de usuário e se utiliza do *back-end* para executar as rotinas dos métodos de retificação. Adotou-

se o *angular* para o desenvolvimento, integrando assim o conceito de *SPAs* ao projeto e dando à página *web* maior responsividade e desempenho, pois a *view* está em *client-side* e não precisa ser carregada pelo servidor.

O uso do *TypeScript* para a programação da lógica da aplicação tornou o desenvolvimento mais ágil e produtivo devido à sua característica multiparadigma. Por um lado, a orientação a objetos permite a reutilização de código e simplifica a integração dos componentes do sistema, auxiliando na manutenção dos requisitos do projeto (KAMATH; SMILAN, SMITH., 1993, 14-24). Já a orientação a eventos facilitou a codificação de métodos de resposta para a ação do usuário no *layout* do programa, como o *click* de botões e a interação com telas de *pop-up*.

O uso da biblioteca *angular material* permitiu a criação e customização dos componentes visuais de maneira simplificada e produtiva, dando maior flexibilidade nas etapas de programação e estabelecendo uma identidade visual para o projeto. A aplicação apresenta um formulário, que pode ser visto na figura 1, composto de *radio-buttons* para definir os parâmetros de método de retificação escolhido: tamanho da janela; formato da janela; filtros de retificação e número de iterações.

Figura 1 – Formulário da aplicação *web* de retificação

Fonte: Autoria própria (2020).

Outra funcionalidade desenvolvida é o sistema de importação de arquivos *.txt* e *.csv* utilizado para realizar a leitura dos pontos amostrais de uma zona de manejo, composto por par de coordenadas e o valor de classe.

Todos os parâmetros selecionados no formulário, bem como os dados lidos do arquivo *.txt* ou *.csv* são enviados ao *back-end* através do protocolo *HTTP*. Os dados são manipulados nas rotinas (*back-end*) e o resultado retorna ao usuário do sistema, apresentado em forma de mapa temático e exibido em um novo componente.

No mapa temático pontos amostrais são simbolizados por meio de círculos e a cor representa a classe ou Zona de Manejo ao qual pertencem, relação esta ilustrada ao usuário através de uma legenda. Para facilitar a visualização foram adicionados recursos nativos do *Openlayers*: botões de *Zoom in* e *Zoom out*, uma linha de escala dinâmica e também um botão para visualização em tela cheia.

As cores das classes podem ser personalizadas na forma de degrade, geralmente utilizado para simbolizar escalas de valor para uma única grandeza, como quantidade de colheita, minerais ou até irrigação.

No componente de exibição, além do mapa, foram implementados recursos personalizados em menus *dropdown* para auxiliar o usuário no acesso e na interpretação dos dados. O primeiro, composto de *radio-buttons*, permite alternar entre pontos amostrais antes ou depois da retificação, facilitando na identificação de mudanças na organização das Zonas de Manejo.

Como exemplificado na figura 2, uma segunda funcionalidade é a de obter as informações de cada ponto selecionando-o por meio de um clique, na qual sua cor é alterada para simbolizar que ele foi selecionado, e então sua longitude, latitude, classe original e classe após retificação estarão disponíveis para consulta no segundo menu lateral.

Figura 2 – Apresentação dos dados de um ponto amostral e da legenda do mapa

The screenshot shows a web application interface. On the left, there is a 'Dados das Layers' panel with a 'Seleção de Layers' dropdown. Below it is a 'Ponto Amostral' section with input fields for Latitude (-54.00608645223), Longitude (-25.40768132128), Classe (2), and Classe Após Retificação (2). A 'Cor do seletor' button is below these fields. At the bottom of this panel is a 'Download Mapa JPG' button. On the right, there is another 'Dados das Layers' panel with a 'Seleção de Layers' dropdown. Below it is a 'Legenda do Mapa' section with a table showing five classes with corresponding colors (blue, yellow, red, green, dark red) and a 'Degradê' button. At the bottom of this panel is a 'Download Mapa JPG' button. In the center, a map displays a cluster of colored pixels representing different classes, with a white line indicating a boundary or path.

Fonte: Autoria própria (2020).

Também foi criado um módulo de tutorial contendo conceitos teóricos relacionados a área agrícola como, definições de agricultura, zona de manejo, *layers* e retificação. Bem como foi desenvolvido um guia passo a passo ilustrando como preencher o formulário e importar os arquivos das Zonas de Manejo, com o objetivo de auxiliar os usuários em relação à utilização do aplicativo computacional.

Foi implementado ainda, um módulo que possibilita a visualização e edição dos dados importados dos arquivos. Desta maneira, é possível alterar os valores de classe antes de executar o método de retificação, bem como, o usuário tem a

opção de criar um novo arquivo com nome e extensão (.txt ou .csv) personalizáveis, mantendo assim uma cópia com os valores originais.

Ao final do ciclo de desenvolvimento o código fonte da aplicação foi refatorado, buscando pôr em prática os conceitos de serviços e componentes, provenientes do *angular*, bem como modularização e desacoplamento. Esta reorganização dos arquivos e otimização das funcionalidades implementadas conferiu ao projeto maior legibilidade, fator importante caso haja necessidade de manutenção ou expansão.

CONCLUSÃO

Um dos aspectos mais importantes deste projeto é que todos os conceitos técnicos e funcionais do método original implementado (BETZEK et al., 2018) foram mantidos intactos, garantindo assim os mesmos resultados obtidos no estudo desenvolvido anteriormente. Entretanto, com as mudanças na camada *view* da aplicação, várias melhorias, tanto na entrada quanto na saída de dados foram alcançadas.

No trabalho original a inserção de Zonas de Manejo e a apresentação de mapas temáticos não eram funcionalidades nativas, dependendo assim da instalação de outro software para realizá-las. Os parâmetros do método tinham de ser inseridos diretamente no código fonte, dificultando o processo e criando falhas no quesito de segurança.

Na proposta atual, foram implementados módulos para a inserção de Zonas de Manejo e apresentação de mapas temáticos, isto, aliado a uma arquitetura *web* que requer do usuário apenas um navegador e conexão com a internet para funcionar, atribuiu simplicidade e praticidade ao uso do programa.

A inserção de parâmetros agora é feita a partir de interfaces desenvolvidas com foco na usabilidade e experiência do usuário, de forma que até mesmo indivíduos menos experientes com ferramenta computacionais possam utilizá-las. Característica esta, ainda mais ressaltada pela implementação do módulo de tutorial, empregado tanto para sanar possíveis dúvidas teóricas sobre agricultura e retificação, bem como no preenchimento do formulário principal.

Com estes aprimoramentos obtidos com a adoção de um *front-end* e de uma arquitetura de microsserviços, todo o desenvolvimento proposto anteriormente (BETZEK et al., 2018) agora está acessível para públicos mais diversos, como agricultores, engenheiros, pesquisadores e estudantes, não se fazendo mais necessário conhecimento profundo sobre áreas da tecnologia da informação.

AGRADECIMENTOS

Agradeço a UTFPR, à Fundação Araucária e ao CNPQ por possibilitarem a pesquisa e desenvolvimento do trabalho e agradeço especialmente ao meu orientador pelo suporte e atenção dada durante todas as etapas do projeto.

REFERÊNCIAS

BETZEK, N. M. et al. Rectification methods for optimization of management zones. *Computers and Electronics in Agriculture*, v. 146., p. 1-11, mar. 2018.

HARRIS, R. D.; FULLER, Q. D. Agriculture: Definition and Overview. In: Encyclopedia of Global Archaeology, 1. ed. New York: Springer, 2014. p. 104-113.

HAZZARD, E. **OpenLayers 2.10 Beginner's Guide**. 1 ed. Olton: Packt Publishing, 2011.

KAMATH, Y. H.; SMILAN, R. E.; SMITH, J. G. Reaping benefits with object-oriented technology. AT&T Technical Journal, vol. 72., n. 5, p. 14-24, set./out. 1993.

MEYER, E. A. **Cascading Style Sheets: The Definitive Guide**. 2 ed. Sebastopol: O'Reilly Media, 2004.

OLIVEIRA, D. J. **Uma Proposta de Arquitetura para Single-Page Applications**. 2017. Tese de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade Federal de Pernambuco. Recife, 2017. Disponível em: <https://www.cin.ufpe.br/~tg/2017-2/dio-tg.pdf> . Acesso em: 05 de maio 2020.

SOBJAK, R. et al. REDUNDANT VARIABLES AND THE QUALITY OF MANAGEMENT ZONES. Journal of the Brazilian Association of Agricultural Engineering, v. 36., p. 78-93, jan./fev. 2013.