CBIS'24
XX Congresso Brasileiro de Informática em Saúde
08/10 a 11/10 de 2024 - Belo Horizonte/MG - Brasil

API REST para reconhecimento óptico de caracteres em rótulos alimentares

A REST API for optical character recognition in food labels

API REST para reconocimiento óptico de caracteres en etiquetas de alimentos

Gabriel Menin[1], Renan Augusto Pereira[2], Flávia Magalhães Guedes[2], Ana Trindade Winck[3]

1 Bel, Federal University of Health Sciences of Porto Alegre - UFCSPA, Porto Alegre (RS), Brazil.
2 Me, Federal University of Health Sciences of Porto Alegre - UFCSPA, Porto Alegre (RS), Brazil.
2 Dra, Federal University of Health Sciences of Porto Alegre - UFCSPA, Porto Alegre (RS), Brazil.

Corresponding author: Profa. Dra. Ana Trindade Winck
E-mail: anatw@ufcspa.edu.br

Resumo

Objetivo: A prevalência de alergias alimentares é uma ameaça à saúde pública, especialmente quando alérgenos são consumidos inadvertidamente. Este estudo desenvolveu uma Interface de Programação de Aplicações para extrair informações sobre ingredientes de rótulos alimentares para, quando integrada a um aplicativo, identificar alérgenos e notificar aos usuários, permitindo que façam escolhas alimentares informadas. Método: Utilizou-se uma biblioteca de reconhecimento ótico de caracteres, calibrada para leitura e tradução de textos em rótulos alimentares. Resultados: Testes com 76 produtos alimentares rotulados, divididos em 7 tipos de materiais, avaliaram a similaridade entre rótulos reais e transcrições realizadas, com uma taxa de similaridade média de 81,61% Conclusão: A solução se mostra viável para integração com um aplicativo de reconhecimento de alérgenos, embora a transcrição automática dos rótulos se mostre mais favorável a determinados tipos de materiais e formas de embalagens de alimentos, sendo necessário uma melhor calibragem para os demais.

Descritores: Rotulagem de Alimentos; Hipersensibilidade Alimentar; Processamento de Imagem Assistida por Computador.

Abstract

Objective: The global prevalence of food allergies is a public health threat, particularly when allergens are inadvertently consumed. This study aims to develop an Application Programming Interface to extract ingredients information from food labels, which, when integrated into an application, can identify allergens and notify users, enabling them to make informed dietary choices. Method: This work applies an optical character recognition library, calibrated for reading and translating texts from food labels. Results: We performed tests with 76 labeled food products, divided into 7 types of materials, evaluating the similarity between actual labels and the transcriptions, achieving an average similarity rate of 81.61%. Conclusion: The solution proves viable for integration with an allergen recognition application, although the automatic transcription of labels is more favorable for certain types of materials and shapes of food packaging, requiring better calibration for others.

Keywords: Food Labeling; Food Hypersensitivity; Image Processing, Computer-Assisted.

Resumen

Objetivo: La prevalencia de las alergias alimentarias es una amenaza para la salud pública, especialmente cuando los alérgenos se consumen inadvertidamente. Este estudio busca desarrollar una Interfaz de Programación de Aplicaciones para extraer ingredientes de etiquetas alimentarias para, al integrarse a una aplicación, identificar alérgenos y notificar a los usuarios, permitiéndoles tomar decisiones alimentarias informadas. Método: Se utilizó una biblioteca de reconocimiento óptico de caracteres, calibrada para leer y traducir textos en etiquetas alimentarias. Resultados: Las pruebas con 76 productos alimenticios etiquetados, divididos en 7 tipos de materiales, evaluaron la similitud entre etiquetas reales y transcripciones, con una tasa de similitud promedio del 81,61%. Conclusión: La solución se muestra viable para la integración con una aplicación de reconocimiento de alérgenos, aunque la transcripción automática de las etiquetas resulta más favorable para ciertos tipos de materiales y formas de envases de alimentos, siendo necesaria una mejor calibración para los demás.

Descriptores: Etiquetado de Alimentos; Hipersensibilidad a los Alimentos; Procesamiento de Imagen Asistido por Computador.

Introduction

Food allergies constitute a public health issue with the potential for causing serious harm, including life-threatening risks. Recent statistics indicate a global prevalence of approximately 10% in children and adolescents, with a notable increase in recent decades[1]. In the Brazilian context, 6 to 8% of children under 3 years old and 2 to 3% of adults experience allergic reactions linked to food[2].

The predominant treatment for food allergies involves the exclusion of the identified allergen or one possibly associated with symptoms, necessitating the removal of the allergen from derived products and preparations containing it to avoid cross-contact. The success of this treatment depends on the appropriate exclusion of the allergen and the ability to meticulously and accurately examine food labels to be consumed[3].

Through labeling, consumers should have access to accurate information about the characteristics of the foods they consume. However, such issues persist, even despite the mandatory inclusion of nutritional information on packaged foods in Brazil. The complexity of language, font sizes in descriptions, coupled with the lack of an unified standard for allergen nomenclature, can lead to health complications, especially for individuals with food allergies, facing the risk of accidental ingestion of allergenic foods[2].

In response to these challenges, technological approaches, such as mobile applications, emerge as solutions to promote accessibility in reading food labels[4]. These resources offer alternatives to overcome obstacles by providing users with tools capable of handling difficulties in identifying allergens on food labels. In this context, technologies that perform text recognition in images play an important role, acting as aids in identifying information on food labels with complex patterns.

By reviewing the literature, we searched for studies focusing on the need for means to identify ingredients on the labels of packaged foods, assisting individuals with dietary restrictions. Some work explore the content of food labels by reading the barcode to communicate with a kind of computational service that maintains the label data; by the other hand, others use Optical Character Recognition (OCR). OCR is a technology that converts images or physical documents into searchable and editable text, thereby facilitating the efficient digitization and management of extensive volumes of text-based data[5]. Since not all available labels may be stored or updated in a single database, we believe that directly extracting text from labels, instead of reading their barcode, is an option that can reach a larger number of food products. Therefore, from the found studies, we selected those that explore OCR in the context of food or nutrition.

Johan and Rizal[(6)] presented a mobile application designed for Android that incorporates OCR and the Boyer Moore algorithm to discern allergens in food products. The application, utilizing Firebase as its database, offers allergen-related information to prevent adverse reactions in individuals with food allergies. The study reports an accuracy surpassing 90% across various test scenarios.

Matsunaga and Sullivan[(7)] contributed to the field by developing a programming interface utilizing computer vision for nutritional data analysis and extraction. Their objective was to streamline user data input and diminish manual efforts. Their approach involves a C++ library amalgamating image pre-processing, OCR, and post-processing, facilitating the direct extraction of nutritional information from food labels. The study discerned an average accuracy rate of approximately 80% for the majority of analyzed label images.

Cakić et al.[(8)] article also addresses the use of computer vision, but for reading serial numbers on wine labels, aiming to develop applications for individual bottle tracking and counterfeiting prevention. Employing OCR techniques and the Tesseract library in conjunction with Python, the authors reported an initial recognition rate of approximately 62%, which subsequently increased to 87.5% following the application of image pre-processing techniques.

The reviewed studies collectively underscore the multifaceted applications of OCR within the domain of food and nutrition. Johan and Rizal's[(6)] mobile application not only caters to individuals with food allergies but also exemplifies the potential of OCR in enhancing food safety, on achieving a remarkably high accuracy rate. While their approach reports an average accuracy rate of approximately 80%, the Matsunaga and Sullivan[(7)] final system proposed still has some room for future accuracy improvement - as the authors themselves mention in their conclusion - such as exploring more OCR options to obtain better results. Although the problem of detecting numbers from images is not solved for all cases, Cakić et al.[8] exploration of OCR for reading serial numbers on wine labels offers a noteworthy recognition rate of 87.5%. In considering these advancements, it is evident that OCR holds substantial promise in diverse applications within the food and beverage industry.

All the studies highlight the ongoing challenges, particularly in achieving consistently high accuracy rates across diverse scenarios and label formats, emphasizing the need for ongoing research and refinement in this field. However, it is less common to find work that simultaneously addresses OCR applied on food labels in the context of food or nutrition, developed over an interface allowing communication with software applications.

In light of these considerations, we developed an REST API, that is, an Application Programming Interface (API) following the Representational State Transfer (REST) architecture, in Java, employing OCR techniques to recognize and digitize text from images of food labels, available over different kinds of packaging or label material. The proposed API receives an image food label and returns the ingredients listed on it. Consequently, when integrated into an application, this REST API provides information used to identify the presence of allergens, aiming to assist patients with food allergies in making informed decisions regarding their dietary choices.

Methods

The methodology adopted for the development of this REST API comprises five main stages: i) a Feasibility Study and Selection of the OCR Library; ii) a NoSQL database modeling for the REST API test; iii) the development environment setup; iv) the REST API development; and v) tests.

Feasibility Study and Selection of the OCR Library

With the aim of establishing an OCR API REST, a preliminary analysis was imperative to discern and assess potential OCR tools and technologies. The primary goal was to identify a library possessing the requisite attributes for constructing the REST API, taking into consideration the contextual aspects inherent to the application. The selection process necessitated an examination to ensure that the chosen tool contains the following characteristics:

- Recognition of special characters and symbols
- Comprehensive and detailed documentation
- Continuous library updates
- Effective Integration with Java Programming Language

Based on this analysis, three options seemed promising at first: GNU Ocrad, an open source OCR library based on a feature extraction method; Asprise Java OCR, a library with functionality for extracting text and barcode information from scanned documents; and Tesseract OCR, an open source OCR library available under the Apache 2.0 license.

The Asprise Java OCR library, while offering a free version for testing, imposes charges for its actual usage; hence, it was promptly excluded. The GNU Ocrad did not exhibit significant recognition capabilities with the tested label images. Consequently, the chosen library was Tesseract OCR, a decision also grounded by its use in the work of Cakić et al.(8) and the positive outcomes achieved in OCR.

NoSQL database modeling

At first, we proposed to model a NoSQL database to test and validate the API-REST, focusing on the OCR model developed. We chose MongoDB(9) due to its flexibility and integration with various frameworks.

The developed database stores data for evaluating the accuracy of text transcription performed by the OCR algorithm. This construct takes the form of a data collection, comprising information related to food products. The structure includes the name and label of the product, along with the transcribed content and the similarity between the transcription and the actual label.

Development environment setup

We strategically employed the integration of Spring Boot(10), along with the Java programming language, to reach the objective to construct a REST API. The choice of Spring Boot is due its capacity to speed up project initialization, avoiding manual configuration files, as well as its inclusion of an embedded application server. Furthermore, Spring Boot simplifies database access by autonomously managing project dependencies, adhering to the principles of Inversion of Control and Dependency Injection inherent in the Spring Framework.

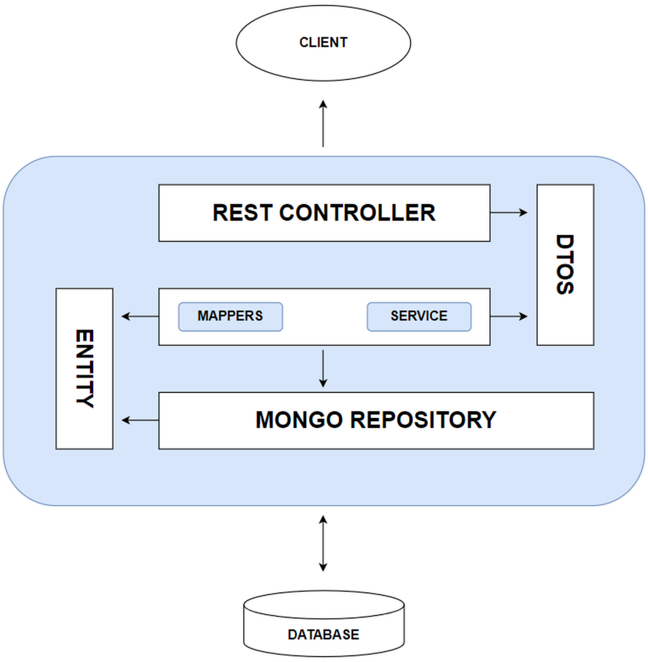
These are the key dependencies Adopted:

- Spring Web: this dependency was selected for its comprehensive support in web application development and adept handling of the HTTP protocol.
- Spring Data MongoDB: chosen for its seamless integration capabilities with MongoDB, facilitating streamlined database operations.
- Tess4J: an integral Java JNA wrapper for the Tesseract OCR library.
- Apache Commons Text and Apache Commons Lang: employed for string manipulation tasks within the project.
- OpenCV: integrated for its utility in the manipulation of image files.
- OpenAPI: utilized for the generation of API documentation.

API Development

In adherence to the Model-View-Controller (MVC) design pattern, the project's organizational architecture was arranged into distinct layers, consisting of the model, repository, service, and “rest” packages. Within the “rest” package, a further subdivision is established, delineating “controller” and “dto” (Data Access Object) layers, as shown in Figure 1. In this Figure, each box represents a layer, while the arrows indicate the direction of communication between them.

Figure 1 – Architecture used in the development of the REST API.



This deliberate structuring facilitates a systematic separation of concerns, ensuring the distinct demarcation of responsibilities within each layer. Such stratification serves to prevent the encroachment of rules from one layer onto another, thereby fostering a modular and comprehensible architectural design.

- Endpoints: the API architecture comprises five distinct endpoints, one of which is dedicated to the execution of OCR, where the Long Short-Term Memory (LSTM) neural network within the Tesseract OCR library undertakes the identification of textual content from a provided image file, subsequently returning the identified text. The remaining four endpoints were designed to facilitate API verification and validation processes. These endpoints operate on the manipulation of domain objects, orchestrating data persistence operations. These operations encompass the registration of a food product along with its corresponding label, the storage of similarity metrics between the actual label and the label transcribed by OCR, as well as the retrieval of all registered products within the test database (or a specific product), and the deletion of a product from the database.

- Text extraction function: this function within the system is entrusted with the task of OCR. This function receives an image as a parameter and returns the identified text through the utilization of a trained model within the Tesseract OCR library.

  In addition to Tesseract, this function leverages the computer vision library OpenCV for digital image pre-processing operations. These operations are designed to accentuate the contours of the letters, thereby enhancing the accuracy in character identification within the image. Among the adopted steps aimed at refining the transcription accuracy is the application of a Gaussian blur, employed to mitigate image noise.
- Similarity comparison function: this function serves the purpose of assessing the agreement between the actual label of a food product and the label transcribed by the text extraction function, yielding the result in the form of a percentage of similarity. This function has been devised specifically for the testing and validation endeavors of the API, incorporating the Jaro-Winkler method[11] to compute the similarity between two strings. In this method, the computed distance ranges from 0 to 1, where a value of 1 signifies maximum similarity, and 0 denotes the absence of resemblance.

Tests

The testing phase of the API should consider a variety of real labels, packaged in different types of materials as well as different formats. The testing phase aims to select food products that meet this variety and capture the ingredients of the selected food labels in two ways:

- Using the developed REST API, storing the interpreted label;
- Using the official website of each food product, storing its real label.

Tests should consider the similarity rate between the ingredients in the real food label and the ingredients in the label captured by the REST API. Finally, there will be computed the average and standard deviation of the similarity rate obtained for each group of foods packaged in a particular type of material

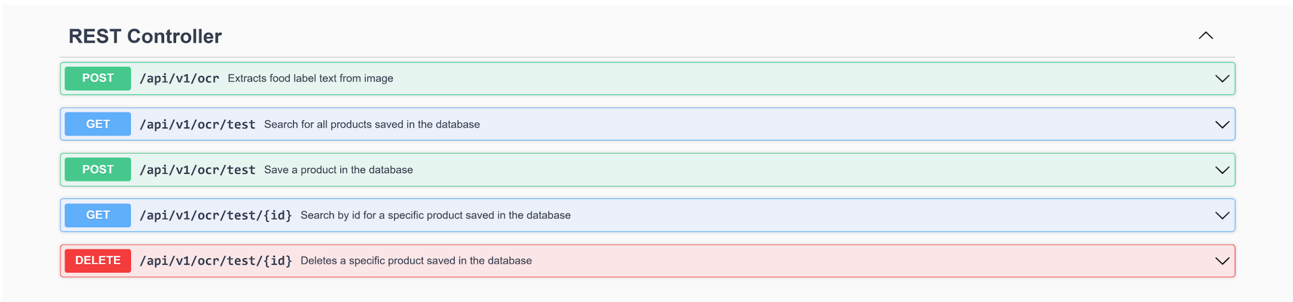
Ethical Aspects

All project stages were conducted using public databases and texts available on the labels of industrialized products. Therefore, in accordance with Resolution No. 510 of April 7, 2016, from the National Health Council, this work does not need to be submitted to an ethics committee.

Results

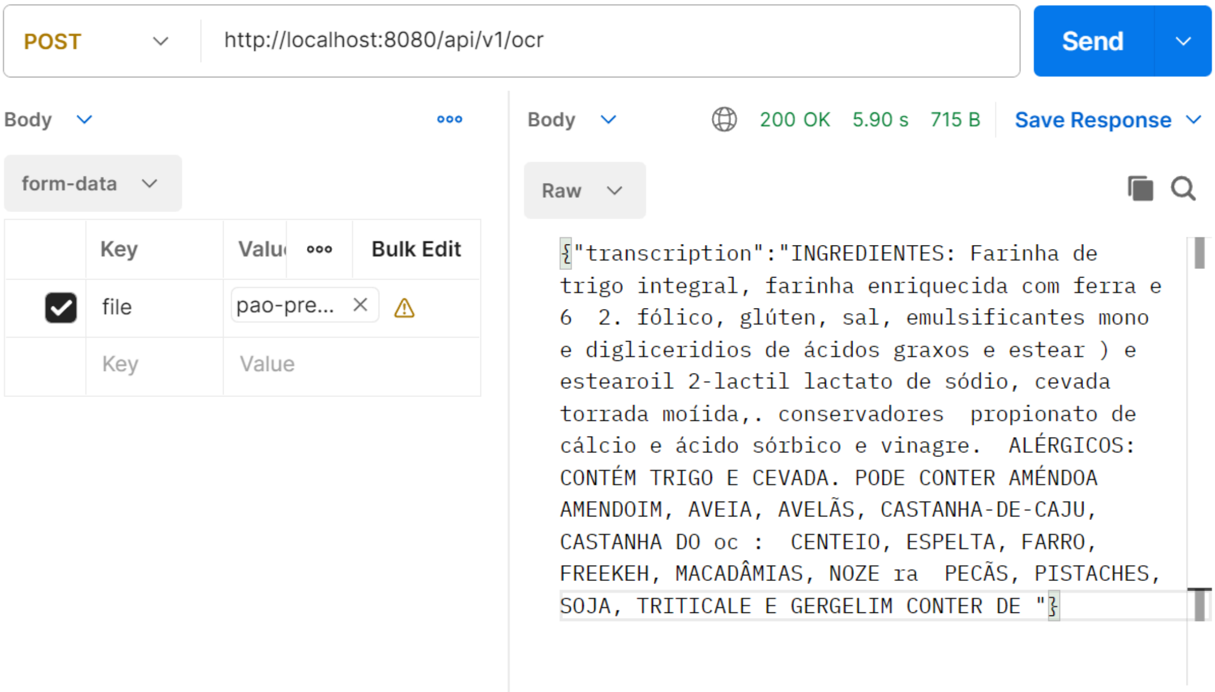
In alignment with the outlined objectives, the structure of the application's operational flow, and the nature of the API's domain object, the five defined endpoints were mapped in the API documentation. The Swagger Framework[12] was employed for this purpose, which enables the specification of the list of available resources in the API and the operations that can be invoked concerning these resources. The visual representation of this documentation process is elucidated in Figure 2.

Figure 2 – The five endpoints mapped in the API documentations, using Swagger Framework



The endpoint tasked with OCR execution invokes the text extraction function from the service layer to process the supplied image as a parameter and transcribe the identified text, subsequently returning the processed result. An illustration of such a request is provided in Figure 3.

Figure 3 – OCR endpoint request example with Postman



The image processing occurs smoothly, as the Tesseract library executes a standardized procedure with parameters and techniques adjustable for each image based on its unique characteristics. Despite the subtle nature of the adjustments made internally by the function, their discernibility is evident, as depicted in Figure 4.

Finally, following the tests phases, we were able to select 76 brazilian food products for testing the API, which are grouped according to packaging or label material. The 76 food products could be divided in 7 different materials, which are detailed in Table 1. It is important to mention that these food products not only present different materials but also distinct shapes. However, avoiding a wide range of combinations, Table 1 does not discriminate packaging shapes. Hence, Table 1 shows the type of packaging/label material, the number of products selected per category and the mean and standard deviation referring to the similarity rate between the actual ingredients food label and the transcripted one.

Figure 4 – Comparative evaluation between the original image and the pre-processed version derived from the label of whole wheat bread "Vital", manufactured and packaged in Low-Density Polyethylene

Table 1 – Similarity rate tested within 76 real products, divided in 7 distinct label materials. Column 1 shows label material; column 2 shows the number of images tested for the respective material; column 3 and for shows, respectively, the similarity mean and standard deviation.

Label Material	No. of Images	Similarity Mean	Standard Deviation
Polypropylene (PP)	14	81.56	12.56
Paper	26	84.80	12.75
Generic Plastic	11	81.26	13.71
Low-density Polyethylene (LDPE)	4	84.39	3.73
Metal	5	77.28	17.18
Biaxially-Oriented Polypropylene (BOPP)	11	84.26	12.08
Polyethylene Terephthalate (PET)	5	77.71	11.76

Conclusion

The Tesseract library has demonstrated notable versatility, exhibiting a spectrum of character identification accuracy, varying from moderate to high, contingent upon the characteristics of the processed images. The library performs optimally in documents featuring uncomplicated layouts. However, challenges arise when confronted with labels characterized by intricate layouts, such as those incorporating stylized fonts or intricate packaging designs. In such instances, to achieve more satisfactory results, it may necessitate supplemental pre-processing or post-processing steps.

The evaluation of the developed REST API reveals a good performance when applied to products featuring flat packaging, exemplified by milk cartons and certain types of cookies and bread packaging. Conversely, challenges manifest in character identification for products enclosed in cylindrical packages, regardless of packaging material as observed in tomato sauce, canned peas, chocolate milk and ice creams.

Material composition also influences performance, with labels on products packaged in materials reflecting more light posing additional complexities due to heightened difficulty in delineating character edges in the image.

Despite these considerations, the API has achieved notably satisfactory results in the domain of OCR within images, particularly evident in the evaluation of similarity between actual labels and those transcribed by the API through the testing endpoint. Future enhancements are envisioned to address identified limitations, such as character recognition in images featuring diverse orientations—a current challenge influenced by constraints in the utilized Java library. Additionally, it is considered important to optimize the OCR process for images of cylindrical labels and images with low contrast, an obstacle currently identified. To achieve this, the proposal involves utilizing the image histogram, a graph representing the distribution of intensities or colors of pixels in an image. This approach allows for the application of image processing techniques such as brightness and contrast adjustment, equalization, and image refinement, employing adaptable parameters for each image, which is deemed pertinent for augmenting result accuracy in subsequent iterations of the project.

References

1. Lobo FATF, Santos MA dos, Montes LTP. Alergia Alimentar: Um Problema Crescente. Saúde em Foco. 2021; 8(3):39–53.

2. Berzuino MB, Fernandes RDC de Sá, Lima MDA, Matias ACG, Pereira IRO. ALERGIA ALIMENTAR E O CENÁRIO REGULATÓRIO NO BRASIL. Revista Eletrônica de Farmácia. 2017 Sep 18;14(2).

3. Solé D, Silva LR, Cocco RR, Ferreira CT, Sarni RO, Oliveira LC, et al. Consenso Brasileiro sobre Alergia Alimentar: 2018 - Parte 2 - Diagnóstico, tratamento e prevenção. Documento conjunto elaborado pela Sociedade Brasileira de Pediatria e Associação Brasileira de Alergia e Imunologia. Arquivos de Asma, Alergia e Imunologia. 2018;2(1).

4. O que é o Desrotulando? | Desrotulando - Central de Ajuda [Internet]. Available from: https://ajuda.desrotulando.com/pt-BR/articles/6301097-o-que-e-o-desrotulando.

5. J. Memon, M. Sami, R. A. Khan and M. Uddin, "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)," in IEEE Access, vol. 8, pp. 142642-142668, 2020

6. Johan EB, Rizal A. Allergen Recognition in Food Ingredients with Computer Vision. Ultima Computing : Jurnal Sistem Komputer. 202;13(2):44–9.

7. Matsunaga N, Sullivan R. Image processing for the extraction of nutritional information from food labels [Internet] [PDF]. [Santa Clara: Santa Clara University]; 2015. Available from: https://scholarcommons.scu.edu/cgi/viewcontent.cgi?article=1041&context=cseng_senior.

8. Cakic S, Popovic T, Sandi S, Krco S, Gazivoda A. The Use of Tesseract OCR Number Recognition for Food Tracking and Tracing. 2020 24th International Conference on Information Technology (IT). Zabljak, Montenegro, 2020, pp. 1-4.

9. MongoDB Documentation [Internet]. https://github.com/mongodb/docs-bi-connector/blob/DOCSP-3279/source/index.txt. Available from: https://www.mongodb.com/docs.

10. Webb P, Syer D, Long J, Nicoll S, Winch R, Wilkinson A, et al. Spring Boot Reference Documentation [Internet]. Available from: https://docs.spring.io/spring-boot/docs/3.1.5/reference/pdf/spring-boot-reference.pdf.

11. Wang Y, Qin J, Wang W. Efficient Approximate Entity Matching Using Jaro-Winkler Distance. In Web Information Systems Engineering – WISE 2017: 18th International Conference, Puschino, Russia, October 7-11, 2017, Proceedings, Part I. Springer-Verlag, Berlin, Heidelberg, 231–239.

12. Get Started With The OpenAPI Specification [Internet]. Swagger.io. 2021. . Available from: https://swagger.io/solutions/getting-started-with-oas/.