

UNIVERSIDADE FEDERAL DO PAMPA

DEISE DALL'AGNOL SEVERO

**BELEZATI: DESENVOLVIMENTO DE UM MÓDULO DE ESCANEAMENTO E
DESCRIÇÃO DE INGREDIENTES COSMÉTICOS BASEADO NA WEB
UTILIZANDO TECNOLOGIA OCR VIA API**

Porto Alegre

2024

DEISE DALL'AGNOL SEVERO

**BELEZATI: DESENVOLVIMENTO DE UM MÓDULO DE ESCANEAMENTO E
DESCRIÇÃO DE INGREDIENTES COSMÉTICOS BASEADO NA WEB
UTILIZANDO TECNOLOGIA OCR VIA API**

Dissertação apresentada ao Programa de Pós-graduação Stricto Sensu em Engenharia de Software da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Mestre em Engenharia de Software.

Orientador: Claudio Schepke

Porto Alegre

2024

Ficha catalográfica elaborada automaticamente com os dados fornecidos
pelo(a) autor(a) através do Módulo de Biblioteca do
Sistema GURI (Gestão Unificada de Recursos Institucionais).

D144 Dall'Agnol Severo, Deise

Belezati: Desenvolvimento de um módulo de escaneamento e descrição de
ingredientes cosméticos baseado na web utilizando tecnologia OCR via API /
Deise Dall'Agnol Severo. – 2024.

59 p. : il.

Orientador: Claudio Schepke

Dissertação (Mestrado) – Universidade Federal do Pampa, MESTRADO EM
ENGENHARIA DE SOFTWARE, Campus Alegrete, 2024.

1. Angular. 2. Cosméticos. 3. OCR. 4. API. 5. Escaneamento. I. Título.

DEISE DALL'AGNOL SEVERO

**BELEZATI: DESENVOLVIMENTO DE UM MÓDULO DE ESCANEAMENTO E
DESCRIÇÃO DE INGREDIENTES COSMÉTICOS BASEADO NA WEB UTILIZANDO
TECNOLOGIA OCR VIA API**

Dissertação apresentada ao Programa de Engenharia de Software da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Mestre em Engenharia de Software.

Dissertação defendida e aprovada em: 19 de janeiro de 2024

Banca examinadora:

Prof. Dr. Claudio Schepke
Orientador
Unipampa

Prof. Dr. Diego Luiz Kreutz

Unipampa

Prof. Dr. Dalvan Jair Griebler
PUCRS



Assinado eletronicamente por **CLAUDIO SCHEPKE, PROFESSOR DO MAGISTERIO SUPERIOR**, em 19/01/2024, às 14:53, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **DIEGO LUIS KREUTZ, PROFESSOR DO MAGISTERIO SUPERIOR**, em 19/01/2024, às 15:31, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **Dalvan Jair Griebler, Usuário Externo**, em 19/01/2024, às 17:19, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



A autenticidade deste documento pode ser conferida no site https://sei.unipampa.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1354625** e o código CRC **3F870184**.

Dedico este trabalho ao meu marido Fernando, que sempre está ao meu lado e sem o seu apoio este estudo não seria possível. Dedico também ao meu filho amado Luiz Fernando que me permite ver o mundo com os olhos de uma criança.

AGRADECIMENTOS

Ao Prof. Dr. Claudio Schepke pela companhia, amizade, incentivo e apoio em todos os momentos que foram necessários para a realização deste estudo, que sempre esteve disponível inclusive em horários além da sua jornada de trabalho para poder me apoiar quando eu precisava.

Aos professores do Programa de Pós-graduação Stricto Sensu em Engenharia de Software da UNIPAMPA por todo o conhecimento compartilhado que contribuíram para minha aprendizagem durante o curso.

Aos amigos e colegas do curso pelas trocas de ideias e experiências que auxiliaram na caminhada desta jornada de estudos e pesquisas.

E a UNIPAMPA que mesmo em meio a uma pandemia não mediu esforços para oferecer a infraestrutura e o ambiente necessários para a realização deste trabalho.

“Se eu vi mais longe, foi por estar sobre os ombros de gigantes”.

Isaac Newton

RESUMO

Um *survey* realizado neste trabalho com 254 participantes revelou que o preço elevado de um produto cosmético é um fator determinante para o participante não comprá-lo. Mas ao saber os benefícios que este mesmo produto pode proporcionar, o participante poderia mudar de ideia e adquiri-lo. O que podemos concluir é que o valor de um produto cosmético não está relacionado ao seu preço e sim ao seu custo-benefício que o mesmo proporciona. Neste sentido foi pensado numa maneira de auxiliar os consumidores a fazerem melhores escolhas no momento da compra de cosméticos, não mais escolhendo pelo preço do produto e sim pelo benefício que o mesmo pode proporcionar. Desta forma, foi desenvolvido um módulo dentro do sistema web existente no mercado, chamado *Belezati*, para informar aos consumidores a função dos ingredientes contidos na composição de um cosmético. Um meio para oferecer esta funcionalidade de forma prática no momento da compra de um produto é a possibilidade de escanear os ingredientes da embalagem do cosmético e no mesmo instante receber informações acerca do produto. Para isto, foram implementadas duas ferramentas de reconhecimento óptico de caracteres disponíveis no mercado por meio de APIs. As ferramentas escolhidas foram a Tesseract e a Amazon Textract. Após as análises, foi concluído que a ferramenta Amazon Textract possui uma performance superior ao Tesseract, que necessita de um tratamento prévio na imagem para a extração do texto. Outro fator de sucesso para as extrações é a resolução da imagem que, quanto maior, melhor é a nitidez dos caracteres, facilitando a extração pelas ferramentas. Além da performance percebida, durante os experimentos ficou evidente que as ferramentas que utilizam reconhecimento óptico de caracteres auxiliam pessoas com baixa visão a compreenderem o que está escrito nos rótulos dos cosméticos, que muitas vezes encontram-se com caracteres muito pequenos para serem lidos a olho nu.

Palavras-Chave: Angular, Cosméticos, OCR, API, Tesseract, Textract, Escaneamento.

ABSTRACT

A survey carried out in this work with 254 participants revealed that the high price of a cosmetic product is a determining factor in the participant not purchasing it. But, the participant could change their mind and purchase it upon learning the product benefits can provide. We can conclude that the value of a cosmetic product is not related to its prices to the cost-benefit it provide. In this sense, we designed a way to help consumers make better choices when purchasing cosmetics, no longer choosing based on the product price but preferably based on the benefits it can provide. In this way, we develop a module within the existing web system on the market, called Belezati, to inform consumers about the function of the ingredients contained in the composition of a cosmetic. One way to offer this functionality in a practical way when purchasing a product would be the possibility of scanning the ingredients on the cosmetic packaging and instantly receiving information about the product. For this, we implement two optical character recognition tools available on the market through APIs. The tools chosen were Tesseract and Amazon Textract. After the analysis, we conclude that the Amazon Textract tool has a higher performance than Tesseract, which requires prior image processing to extract the text. Another success factor for extractions is the image resolution, which, the higher it is, the better the sharpness of the characters, making extraction easier by the tools. In addition to perceived performance, during the experiments it became clear that tools that use optical character recognition help people with low vision understand what on cosmetic labels is written, which often contain characters too small to be read by eye.

Keywords: Angular, cosmetics, OCR, API, Tesseract, Textract, scanner.

LISTA DE FIGURAS

Figura 1 – Etapas de um sistema de Visão Computacional	17
Figura 2 – Exemplo de aplicações clientes consumindo uma API	19
Figura 3 – Diagrama do fluxo de um módulo Angular	22
Figura 4 – Exemplo da arquitetura de uma aplicação Angular	22
Figura 5 – Exemplo de diferentes resoluções de imagem	26
Figura 6 – Etapas do processo de pesquisa deste estudo de caso	30
Figura 7 – Função de cálculo de Acurácia	32
Figura 8 – Código do cálculo de performance	33
Figura 9 – Estrutura do módulo de escaneamento de ingredientes cosméticos	35
Figura 10 – Exemplo de busca pelos ingredientes no dataset	38
Figura 11 – Shampoo em embalagem branca	39
Figura 12 – Shampoo em embalagem preta	41
Figura 13 – Shampoo para bebê	43
Figura 14 – Hidratante para os olhos	44
Figura 15 – Embalagem virada para esquerda	46
Figura 16 – Embalagem virada para direita	47
Figura 17 – Embalagem posicionada em 90°	48
Figura 18 – Embalagem posicionada em 180°	49

LISTA DE QUADROS

Quadro 1 – Trabalhos relacionados	28
Quadro 2 – Análise embalagem clara e caracteres pretos	40
Quadro 3 – Análise embalagem escura e caracteres brancos	41
Quadro 4 – Análise embalagem colorida e caracteres coloridos	43
Quadro 5 – Análise embalagem com baixo contraste e resolução	45
Quadro 6 – Análise embalagem virada para esquerda	46
Quadro 7 – Análise embalagem virada para direita	47
Quadro 8 – Análise embalagem posicionada em 90°	48
Quadro 9 – Análise embalagem posicionada em 180°	49
Quadro 10 – Caracteres comumente trocados	50

LISTA DE SIGLAS E ABREVIATURAS

ABIHPEC	Associação Brasileira de Higiene Pessoal, Perfumaria e Cosméticos
ANVISA	Agência Nacional de Vigilância Sanitária
API	<i>Application Programming Interface</i>
AWS	Amazon Web Services
CLI	<i>Command-line Interface</i>
DPI	<i>Dots Per Inch</i>
GUI	<i>Graphical User Interface</i>
HPPC	Higiene Pessoal, Perfumaria e Cosméticos
HTTP	<i>Hypertext Transfer Protocol</i>
HP	Hewlett-Packard
IDE	<i>Integrated Development Environment</i>
INCI	<i>International Nomenclature of Cosmetic Ingredients</i>
IOS	<i>iPhone Operating System</i>
JSON	<i>JavaScript Object Notation</i>
ML	<i>Machine Learning</i>
MPA	<i>Multi Page Application</i>
MVC	Model-View-Controller
NPM	<i>Node Package Manager</i>
OCR	<i>Optical Character Recognize</i>
PPI	<i>Pixels Per Inch</i>
REST	<i>Representational State Transfer</i>
SPA	<i>Single Page Application</i>
TI	<i>Tecnologia da Informação</i>
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 Objetivos.....	15
1.2 Justificativa.....	15
1.3 Organização do trabalho.....	16
2 FUNDAMENTAÇÃO TEÓRICA.....	17
2.1 Visão computacional.....	17
2.2 APIs e bibliotecas para OCR.....	18
2.2.1 Tesseract x Tesseract.js.....	19
2.2.2 Amazon Textract.....	20
2.3 Angular.....	21
2.4 JSON.....	23
2.5 Nomenclatura dos ingredientes cosméticos no Brasil.....	24
2.6 Resolução de Imagens.....	25
3 TRABALHOS RELACIONADOS.....	27
4 MÉTODOS E PROCEDIMENTOS DA PESQUISA.....	30
4.1 Planejamento.....	30
4.2 Preparação.....	31
4.3 Implementação.....	32
4.4 Experimentos.....	32
4.5 Limitações do Estudo.....	33
5 ARQUITETURA DO MÓDULO.....	35
5.1 Módulos do sistema.....	35
5.2 Esquema do dataset.....	36
6 EXPERIMENTOS E ANÁLISES.....	38
6.1 Embalagem clara e caracteres escuros.....	39
6.2 Embalagem escura e caracteres claros.....	40
6.3 Embalagem colorida e caracteres coloridos.....	42
6.4 Embalagem com baixo contraste e resolução.....	44
6.5 Embalagem posicionada em ângulos diversos.....	45
6.4 Problemas encontrados.....	50
7 CONSIDERAÇÕES FINAIS.....	51
7.1 Trabalhos Futuros.....	52
7.2 Premiações.....	53
REFERÊNCIAS.....	54
APÊNDICE A - FORMULÁRIO: Tecnologia & Cosméticos.....	55
APÊNDICE B - DISPONIBILIDADE DO SISTEMA BELEZATI E TELAS.....	59
ANEXO A - FUNÇÃO DOS INGREDIENTES COSMÉTICOS.....	61

1 INTRODUÇÃO

De acordo com uma pesquisa prévia sobre consumo de cosméticos (APÊNDICE A), realizada nesta dissertação através de um *survey* com 254 participantes, o preço elevado de um produto cosmético é um fator para o participante não comprá-lo. Por outro lado, também revelou que se o mesmo conhecesse os benefícios dos ingredientes da fórmula deste produto, isto poderia fazer com que o entrevistado mudasse de ideia e o comprasse. O que podemos compreender é que as pessoas deixam de comprar produtos cosméticos de maior custo por não saberem quais benefícios o produto pode proporcionar e não pelo seu preço, já que a percepção de valor está relacionada ao benefício que o produto traz. Embora nem sempre produtos de valor elevado sejam os melhores, o que define a relação custo-benefício de um produto cosmético é a combinação dos seus ativos cosméticos, sendo preferível ingredientes que não ofereçam riscos aos usuários.

Uma forma de manter as pessoas informadas acerca da qualidade e riscos dos ingredientes da composição de um cosmético é por meio de pesquisa na Web, já que atualmente, grande parte da população utiliza um aparelho celular com acesso à Internet. Neste cenário, as aplicações baseadas na Web podem alcançar qualquer pessoa em qualquer lugar, sendo necessário apenas um programa do tipo navegador. Desta forma, um meio prático para disponibilizar uma ferramenta que informe os benefícios de um cosmético para auxiliar na escolha do melhor produto poderia ser através do desenvolvimento de um aplicativo.

O sistema *Belezati*¹ existente no mercado para qualificação de atendimento estético por meio de fichas digitais de anamnese. Neste sentido, o sistema também poderia conter um módulo para auxiliar os consumidores de cosméticos a fazerem melhores escolhas por intermédio da disponibilização da informação da função dos ingredientes. Uma forma bastante prática de oferecer essa funcionalidade seria através da possibilidade do usuário tirar uma foto da fórmula contida na embalagem do produto e receber as informações das funções existentes nele como hidratação, proteção solar ou até mesmo se possui algum ingrediente comedogênico.

A questão que norteia esta pesquisa é se podemos fornecer uma ferramenta acessível para que as pessoas saibam as propriedades dos ingredientes cosméticos, escaneando a lista de ingredientes diretamente da embalagem do produto no momento da compra, podendo assim fazer melhores escolhas de acordo com suas condições financeiras e benefícios pretendidos?

¹ <https://app.belezati.com>

1.1 Objetivos

O objetivo geral desta dissertação é o desenvolvimento do módulo de escaneamento de ingredientes cosméticos no sistema *Belezati* com tecnologias Web, utilizando o reconhecimento óptico de caracteres fornecido por APIs públicas de OCR (*Optical Character Recognition* - Reconhecimento Óptico de Caracteres) para a leitura autônoma da lista de ingredientes cosméticos diretamente da embalagem dos produtos, e identificar os desafios da aquisição de imagens diretamente da câmera do aparelho celular de um usuário comum, a fim de ajudar os consumidores de cosméticos a escolherem produtos que melhor atenda às suas exigências. Para tanto, foram definidos os seguintes objetivos específicos neste estudo:

- A. Criar um dataset de ingredientes cosméticos em JSON (JavaScript Object Notation - Notação de Objetos JavaScript);
- B. Analisar as características e funcionalidades de softwares similares ao módulo de escaneamento de ingredientes do sistema *Belezati* para buscar novas soluções para o desenvolvimento do módulo;
- C. Testar a viabilidade de aplicações com OCR por captura de imagem por celular;
- D. Avaliar a performance das APIs públicas de OCR.

1.2 Justificativa

O setor de Higiene Pessoal, Perfumaria e Cosméticos (HPPC) foi considerado essencial pelo governo federal brasileiro no Decreto nº 10.329/2020 durante a pandemia de COVID-19. Embora revogado pelo Decreto de nº 11.077/2022, o panorama do setor de higiene, perfumaria e cosméticos lançado em janeiro de 2023 pela Associação Brasileira da Indústria de Higiene Pessoal, Perfumaria e Cosméticos (ABIHPEC), coloca o Brasil como o 2º país que mais lança produtos para o setor anualmente, atrás apenas dos Estados Unidos e o 4º país em consumo, ficando atrás de Estados Unidos, China e Japão². De acordo com uma pesquisa da Euromonitor, este consumo representa US\$26,9 bilhões, e representa 4% do PIB brasileiro³, sendo assim um importante setor para a indústria e varejo brasileiro. Estes dados demonstram a relevância e força deste mercado, importante para a economia de diversos países.

²<https://abihpec.org.br/comunicado/como-inovacao-e-tecnologia-estao-moldando-o-setor-de-higiene-pessoal-perfumaria-e-cosmeticos/#:~:text=O%20Brasil%20C3%A9%20o%20%2C2%BA,das%20marcas%20e%20da%20ind%20C3%BAstria.>

³https://abihpec.org.br/site2019/wp-content/uploads/2023/01/Panorama-do-Setor_Atualizado_17.10.23_sem_eXs.pdf

O mercado de HPPC está cada vez mais seletivo quanto a qualidade e procedência dos ingredientes. Este fato tem motivado a indústria cosmética a desenvolver pesquisas com ingredientes mais naturais e a produção de produtos veganos, sem ingredientes cosméticos de origem animal, sendo um diferencial competitivo e uma exigência do público consumidor.

Estudos sobre o uso da tecnologia para fomentar o consumo de cosméticos utilizando inteligência artificial vêm sendo desenvolvidos pela grande indústria cosmética, como o caso do sistema Skin Genius⁴, que analisa a situação da pele com a captura de uma foto. Embora os avanços no uso de análise de imagens da indústria sejam relevantes, grande parte tem o objetivo na venda dos seus próprios produtos. Ademais, esta dissertação desenvolve uma ferramenta independente, promovendo o consumo consciente de produtos cosméticos, de forma a servir como suporte na escolha dos produtos ajudando a escolher algo que satisfaça suas necessidades e expectativas.

1.3 Organização do trabalho

Esta dissertação está organizada em sete capítulos, da seguinte forma. O Capítulo 2 apresenta a fundamentação teórica para a compreensão deste trabalho. No Capítulo 3 são apresentados os trabalhos relacionados. O delineamento da pesquisa é descrito no Capítulo 4. A arquitetura do módulo de escaneamento de ingredientes e sua implementação é mostrada no Capítulo 5. No Capítulo 6 são realizados os experimentos e as análises dos resultados. Por fim, as considerações finais e trabalhos futuros são relatados no Capítulo 7.

⁴ <https://www.loreal.com/en/articles/science-and-technology/skin-genius-loreal-paris/>

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos necessários para a compreensão deste trabalho. Isto inclui a caracterização da Visão Computacional e OCR, a contextualização de uma API, a abordagem de duas ferramentas para extração de textos em imagens e documentos, os aspectos da estrutura de dados utilizada pelo software desenvolvido e as resoluções acerca da nomenclatura de ingredientes cosméticos no Brasil.

2.1 Visão computacional

Segundo Russel e Norvig (2022), a Visão Computacional é uma área da Inteligência Artificial cujo objetivo é extrair informações necessárias para tarefas como manipulação, navegação e reconhecimento de objetos. De acordo com Feltrin (2023) e Barelli (2018), independentemente da linguagem de programação ou tecnologia empregada, a Visão Computacional compartilha um fluxo comum, conforme ilustrado na Figura 1.

Figura 1. Etapas de um sistema de Visão Computacional



Fonte: Elaborado pela autora

Os sistemas de Visão computacional possuem em média 5 etapas:

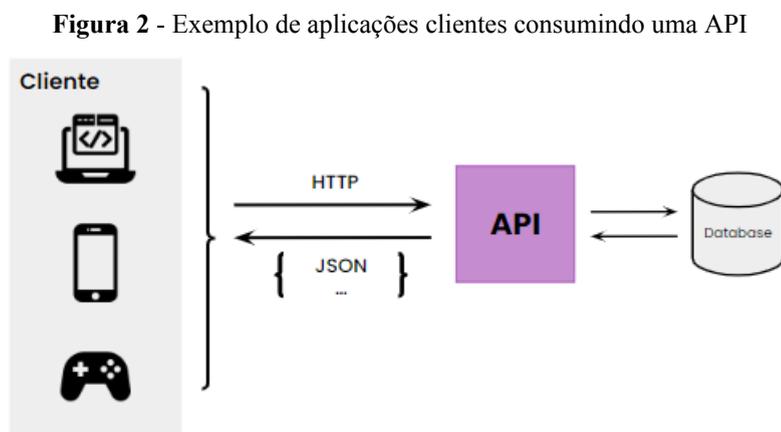
- **Aquisição da imagem:** A primeira etapa de um sistema de visão computacional é a aquisição de imagem ou sequência de imagens, como um vídeo, por meio de sensores, câmeras ou dispositivos de captura de imagens.
- **Pré-processamento:** A segunda etapa compreende o pré-processamento, com o objetivo de melhorar a qualidade da imagem, reduzindo ruídos, iluminação, e distorções, além de utilizar técnicas para destacar bordas ou formas geométricas, a fim de realçar os objetos de interesse para facilitar a obtenção de informações pelo sistema de Visão Computacional.
- **Segmentação:** Nesta etapa os objetos de interesse são separados da imagem original para uma nova imagem, de acordo com Barelli (2018), com o propósito de facilitar a extração das características dos objetos de interesse.
- **Extração de características:** Conforme Feltrin (2023), a extração de características envolve a descrição dos objetos de interesse de uma imagem como bordas, texturas, cores e demais padrões que serão utilizados para representar os objetos para análise.

- **Reconhecimento:** Por fim, a etapa de reconhecimento identifica o objeto segmentado através de suas características e classifica o mesmo com base nas informações fornecidas ao sistema.

A Visão Computacional pode ser aplicada em diversas áreas do conhecimento humano como reconhecimento facial para sistemas de vigilância, reconhecimento de caracteres de placas de veículos para sistemas eletrônicos de estacionamento, automação industrial, auxiliar em diagnósticos médicos por meio de imagens, entre outros. A extração de textos utilizando o reconhecimento óptico de caracteres, mais conhecido pela sua sigla em inglês, OCR (*Optical Character Recognition*), faz parte da área de Visão Computacional. Embora tenha sido melhor explorado nos últimos anos com a evolução e popularização dos dispositivos de captura de imagens, seus estudos datam períodos anteriores ao século XX. Atualmente as pesquisas que envolvem OCR incluem estudos para o melhoramento do reconhecimento dos caracteres utilizando Machine Learning, como forma da máquina aprender a reconhecer padrões nos formatos de letras, inclusive manuscritas e ideogramas.

2.2 APIs e bibliotecas para OCR

API é uma sigla para o termo em inglês *Application Programming Interface*, que segundo Torres (2021), refere-se a um software que fornece um conjunto de definições, regras e protocolos para construir e integrar sistemas, permitindo que uma aplicação se comunique com a outra. É por meio de uma API que aplicações clientes podem requisitar informações e fazer alterações dentro de um servidor do fornecedor. Na Figura 2, temos um exemplo de como aplicações Web, móveis e dispositivos podem consumir informações de um banco de dados através de uma única API, por intermédio de troca de dados no formato JSON, que será descrito na Seção 2.4.



Fonte: Elaborado pela autora.

Uma forma para fornecedores de tecnologia oferecerem seus serviços ao mercado é disponibilizando suas APIs de forma pública. Um exemplo de API Pública é o serviço de reconhecimento óptico de caracteres da Amazon, chamado Textract, utilizado neste trabalho. Nele é feito o envio das informações da imagem através de uma requisição ao servidor do serviço via protocolo HTTP para a API e a mesma retorna os dados, após a imagem ser processada. Outra forma de utilizar recursos de um fornecedor é através de bibliotecas. Essas declaram métodos acessíveis pelo código, que podem ou não incluir uma API dentro de sua arquitetura. Um exemplo de biblioteca é o Tesseract.js, que fornece meios para acessar a API do Tesseract.

2.2.1 Tesseract x Tesseract.js

Tesseract é um mecanismo de reconhecimento óptico de caracteres (OCR) de código aberto escrito em C. Foi originalmente desenvolvido pela Hewlett-Packard (HP) entre os anos de 1985 e 1994, com algumas alterações realizadas em 1996 para rodar no Windows e parte das funcionalidades transcritas para C++ em 1998. Em 2005, o código fonte do Tesseract foi aberto pela HP. De 2006 a novembro de 2018, ele foi desenvolvido pelo Google.

Tesseract, embora não possua uma interface gráfica de usuário (GUI), pode ser utilizado diretamente via linha de comando ou acessado pela sua API desenvolvida em C++, para extrair o texto de imagens. Ele suporta uma variedade de idiomas, inclusive línguas com escritas em ideogramas, como o chinês. Tesseract é um software livre e pode ser utilizado sob os termos da licença Apache 2.0. Para obter um melhor resultado no reconhecimento óptico de caracteres pelo mecanismo do Tesseract, é necessário fazer um pré processamento da imagem, pois o mesmo não possui mecanismos para a melhoria da qualidade da imagem.

Já Tesseract.js é uma biblioteca escrita em javascript, que oferece recursos para a utilização do mecanismo Tesseract. É possível utilizá-la incluindo seu script via a tag `<script>` do HTML ou por meio de seu pacote de arquivos gerado pelo webpack, um empacotador de módulos para javascript, via npm do Node.js, um ambiente de execução de código JavaScript do lado do servidor. NPM é o gerenciador de pacotes padrão do Node.js. Tesseract.js processa a imagem através do envio da localização do arquivo no computador do usuário utilizando a web API FileReader de leitura de arquivos do javascript.

2.2.2 Amazon Textract

Amazon Textract é o serviço de reconhecimento óptico de caracteres (OCR) inteligente da Amazon Web Services também chamada de “AWS”, ele utiliza *machine*

learning (ML) para identificar, compreender e extrair dados específicos de documentos de vários formatos, como arquivos PDFs, imagens, tabelas e formulários, inclusive documentos manuscritos.

Machine Learning em português Aprendizado de Máquina, é uma técnica de Inteligência Artificial que utiliza técnicas estatísticas para permitir que um sistema aprenda com exemplos e melhore o seu desempenho sem ser programado explicitamente. (PRESSMAN e MAXIM, 2021).

Diferentemente de Tesseract, Amazon Textract é capaz de detectar caracteres mesmo em imagens de baixa qualidade, não necessitando de um pré tratamento da mesma, pois o mesmo já faz o pré-processamento da imagem. Além da tradicional detecção de caracteres, ele também permite analisar os documentos processados. Para extrair dados específicos dos documentos ele possui a possibilidade de criar consultas personalizadas em linguagem natural para a extração da informação desejada com base na criação de um dataset de treinamento para a ferramenta por intermédio de um adapter que irá receber as imagens que irão compor o dataset de treinamento e testes.

Amazon Textract possui duas formas de receber os arquivos para o processamento. Uma delas é fazendo o upload do documento para o serviço de armazenamento da AWS - Amazon Simple Storage Service (Amazon S3) e indicando o recipiente onde ele se encontra, conhecido como buckets ou convertendo o arquivo para a codificação em base64. Base64 é um método de codificação de dados para transferência de dados binários na internet. Ele é constituído de 64 caracteres o que deu origem ao seu nome.

Há muitos benefícios na utilização da API do Amazon Textract. Ela permite remover a complexidade da extração de caracteres pelo lado do software cliente, podendo o mesmo focar nas regras de negócio do uso das informações, além de processar os documentos de forma rápida e a baixo custo, pois o serviço é cobrado apenas por documentos processados.

2.3 Angular

Angular é um framework web open source criado pelo Google e escrito em Typescript que permite a criação de aplicações web baseadas no conceito de *Single Page Application* (SPA). Inicialmente lançado como AngularJS em 2010, com o lançamento da sua segunda versão em 2016, foi totalmente redesenhado passando a se chamar apenas de Angular.

Segundo Torres {2021}, uma SPA ou aplicação de página única, é um programa baseado na Web que atualiza, de forma dinâmica as informações de uma página sem a necessidade de recarregar todo o conteúdo que é exibido ao usuário. Isto é diferente de uma *Multi Page Application* (MPA), onde todos os elementos de uma página são renderizados e

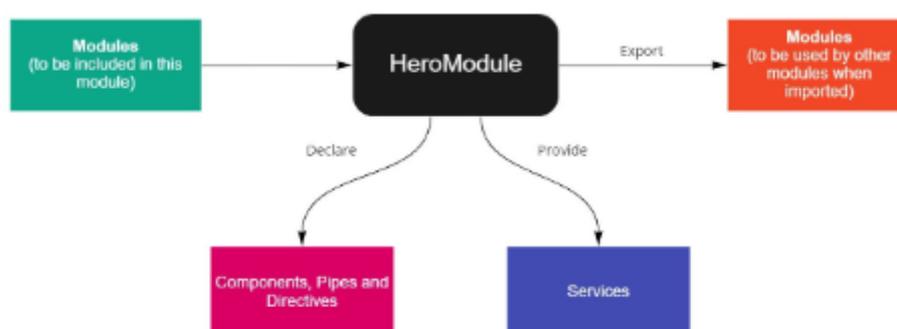
carregados sempre que uma nova informação é solicitada ao servidor, como eram feitas as páginas Web tradicionais. Uma SPA proporciona uma melhor experiência de usuário, pois deixa a navegação fluida, dinâmica e mais rápida, elevando a performance da aplicação. Recarregar todo o conteúdo de uma página web consome tempo e recursos do servidor, além de estar relacionada à velocidade de conexão da Internet de cada usuário, o que libera o tráfego de dados entre servidor/cliente.

A arquitetura do Angular é baseada em componentes que, de acordo com o Duldulao e Villafranca (2022), permite o desenvolvimento da aplicação em partes, como pequenas peças, onde podemos reutilizar essas peças por toda a aplicação, reduzindo a quantidade de código repetidos no projeto, os chamados “*boilerplate code*”.

O Angular possui um módulo principal comumente chamado de AppModule. Dentro dele devem constar todos os componentes da aplicação. Também é possível criar novos módulos dentro do AppModule, a fim de agrupar componentes conforme o domínio da aplicação. Um exemplo que podemos ter é um módulo chamado “template” e dentro dele ter os componentes de *breadcrumb*, rodapé e cabeçalho da aplicação. Ainda podemos ter módulos relacionados ao negócio como de “clientes”, “assinaturas” entre outros. Conforme a aplicação cresce, novos módulos podem ser adicionados, sem prejudicar os módulos existentes, inclusive podendo reutilizar os componentes contidos dentro de outros módulos.

Na Figura 3, pode-se visualizar o fluxo de funcionamento de um módulo no angular, onde o mesmo pode ser exportado para ser utilizado por outros módulos, prover serviços para manipular os dados que serão exibidos nos componentes, bem como declarar componentes, *pipes* e diretivas.

Figura 3 - Diagrama do fluxo de um módulo Angular

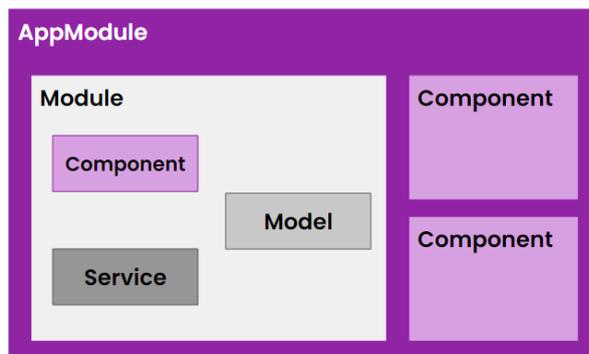


Fonte: Duldulao e Villafranca, 2022.

A Figura 4 ilustra um exemplo de aplicação utilizando Angular como framework. Os componentes não devem acessar diretamente os dados. Isto fica a cargo dos serviços. De

modo geral, os componentes devem focar na exibição dos dados. Podemos utilizar “interfaces” para criar uma “model” como forma de descrever as propriedades de uma entidade, padronizando e facilitando a posterior manutenção da aplicação.

Figura 4 - Exemplo da arquitetura de uma aplicação Angular



Fonte: Elaborado pela autora.

Esse aspecto se assemelha ao padrão de arquitetura de software muito conhecido, chamado MVC (Model-View-Controller), que divide a aplicação em camadas, onde cada camada possui uma responsabilidade única.

O Angular também utiliza o padrão de projeto estrutural *Decorator*, também chamado de *Wrapper*, que adiciona comportamentos a um elemento de forma dinâmica sem ser necessário alterar o código original deste elemento. Geralmente os nomes de *decorators* se iniciam com o símbolo “@”. São exemplos de decorators usados no projeto o `@Component` que define a classe como um componente do sistema e o `@Injectable`, que podemos utilizar nas classes de serviço permitindo a injeção de seus métodos como dependência de um componente.

As responsabilidades primárias de serviços são, segundo Duldulao e Villafranca (2022):

- Reutilizar a lógica em diferentes componentes;
- Implementar comunicação com APIs e acesso a dados;
- Promover a responsabilidade única, separando as funcionalidades dos componentes.

De modo geral, há muitos benefícios em utilizar o framework Angular na criação de aplicações web. Por possuir uma arquitetura robusta que vai além de componentes, serviços e modelos de dados, é possível criar classes enum, guarda de rotas entre outros recursos. Ele também contém um CLI que permite inicializar a aplicação, bem como criar novos recursos diretamente de um terminal. Para iniciar um projeto Angular é necessário ter instalado na máquina o Node.JS, e através do NPM, via terminal, fazer a instalação do Angular CLI.

2.4 JSON

JSON é o acrônimo para o termo *JavaScript Object Notation* (Notação de Objetos JavaScript), um formato para troca de dados entre sistemas ou mesmo tráfego de dados dentro do próprio sistema, criado em 2001 por Douglas Crockford e Chip Morningstar, baseado na notação de objetos literais do JavaScript. JSON ficou muito popular por ser uma alternativa ao XML (Extensible Markup Language, ou Linguagem de Marcação Extensível) tido como mais complexo e menos flexível.

Basset (2019) explica que embora tenha a palavra JavaScript no nome, o JSON é independente de linguagem de programação, não sendo necessário o conhecimento prévio de JavaScript para poder utilizar este formato para troca de dados. Sua sintaxe consiste na escrita de pares de chave-valor, utilizando o caractere de dois pontos, “:”, para separar a chave dos valores. A chave sempre estará à esquerda do separador, e o valor à direita como por exemplo, “nome”：“Deise”. Para criar um arquivo JSON basta salvar o documento com a extensão .json.

Em JSON, utilizam-se os caracteres de chaves “{ }” para indicar um objeto. Logo, também é possível acrescentar outros tipos de dados como string utilizando aspas duplas, numéricos, booleanos, arrays e nulos. Cada par de chave-valor é separado por vírgula, podendo assim acrescentar tantas informações quanto necessárias. O exemplo abaixo mostra um JSON com dados de uma pessoa:

```
{
  "nome": "Deise",
  "idade": 35,
  "ativo": true,
  "calçados": [35, 36],
  "pets": {
    "tipo": "gatos",
    "nomes": ["Luna", "Bella"]
  }
}
```

Basset (2019) complementa que caso o nome da chave seja escrito sem aspas duplas ou com aspas simples, ou mesmo indicar uma *string* com aspas simples, ele será interpretado como um objeto JavaScript, pois o JSON foi baseado nos objetos literais do JavaScript. O uso de aspas duplas é obrigatório para definir o formato de troca de dados como JSON.

2.5 Nomenclatura dos ingredientes cosméticos no Brasil

Segundo a ANVISA, atualmente há disponível mais de 12 mil ingredientes que podem ser utilizados em produtos cosméticos, onde muitos além de sua denominação química, também possuem um nome comercial. Uma forma de padronizar a nomenclatura da lista de

ingredientes nos rótulos dos produtos cosméticos é a utilização do sistema internacional de codificação chamado INCI (International Nomenclature of Cosmetic Ingredients) (ANVISA, 2021), criado em 1973 pela associação americana PCPC (Personal Care Products Council) .

O objetivo do uso da nomenclatura INCI é facilitar a identificação de qualquer ingrediente de forma clara, precisa e imediata não só no Brasil, mas em qualquer país no mundo e principalmente do ponto de vista sanitário, proteger e resguardar a saúde da população (ANVISA, 2021).

Há muitas vantagens na utilização do nome INCI na descrição da composição dos cosméticos, além da facilidade do consumidor identificar o ingrediente em qualquer lugar do mundo, promove uma maior agilidade por órgãos de vigilância sanitária na identificação dos ingredientes, e facilidade para pesquisadores e profissionais na localização das informações na literatura técnico-científica e nos compêndios de referência.

A ANVISA também disponibiliza uma tabela com a função dos ingredientes cosméticos disponível no Anexo I. Além da função do ingrediente, é importante ressaltar que existe uma diferença entre ingrediente vs. substância na indústria cosmética, onde uma substância pode ser a combinação de mais de um ingrediente gerando um novo nome INCI.

A partir de 1º de novembro de 2023 é obrigatório descrever a composição dos ingredientes em português de produtos de higiene pessoal, cosméticos e perfumes, para atendimento da Resolução da Diretoria Colegiada - RDC Nº 646, de 24 de março de 2022 (alterada pela Resolução da Diretoria Colegiada - RDC Nº 773, de 15 de fevereiro de 2023). (ANVISA, 2023).

A resolução - RDC Nº 773 define no Art. 2º parágrafo 4º, a composição química em língua portuguesa deverá ser descrita:

- I. entre parênteses ou após barra, ao lado da respectiva descrição do ingrediente em INCI; ou
- II. Em uma segunda lista, na mesma ordem dos ingredientes em INCI, precedida da expressão "Composição (português):" ou "Ingredientes (português):", podendo ou não estar com todas as letras maiúsculas." (NR).

Para auxiliar a população interessada, a ANVISA desenvolveu um painel, através da ferramenta PowerBI da Microsoft, com a lista de traduções dos ingredientes de produtos de higiene pessoal, cosméticos e perfumes⁵. Outra resolução importante sobre ingredientes cosméticos no Brasil é a RDC Nº 83, de 17 de junho de 2016, que lista substâncias que não

⁵ Disponível em: <http://bit.ly/47jNqBh>

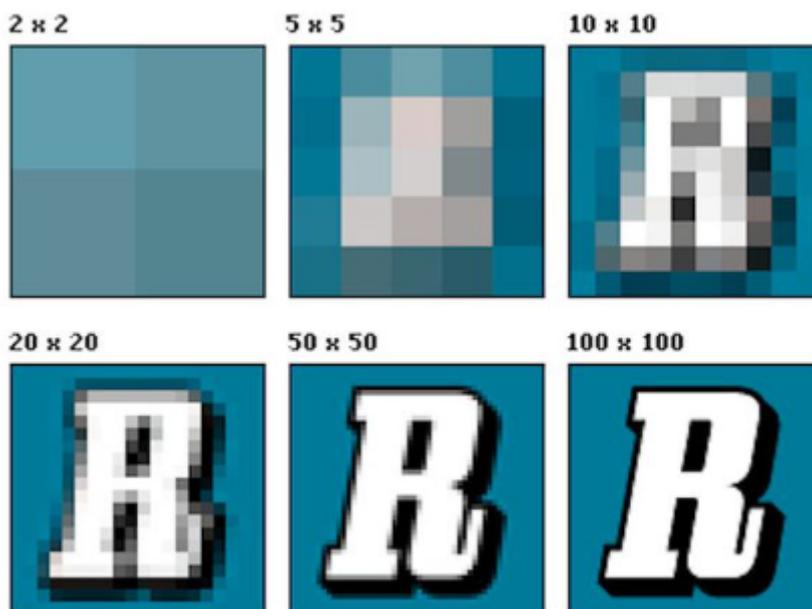
podem ser utilizadas em produtos de higiene pessoal, cosméticos e perfumes. Esta é uma informação importante para a criação de um método para a classificação de ingredientes.

2.6 Resolução de Imagens

A resolução de uma imagem está relacionada à percepção de nitidez da imagem, isto é, com os contornos dos elementos da imagem bem definidos. Freitas (2020) demonstra que a resolução de uma imagem pode ser calculada em DPI (“Dots Per Inch”, pontos por polegada em português) como por PPI (“Pixels Per Inch” ou pixels por polegada em português). Uma imagem digital que possui resolução de 1024 x 768 significa que possui 1024 pixels de largura e 768 pixels de altura. A Figura 5 mostra a mesma imagem em diferentes resoluções.

Dessa forma, esses pixels influenciam diretamente na imagem, à medida que uma imagem com maior resolução, possui uma maior quantidade de detalhes, enquanto uma imagem com uma resolução menor, ficará borrada e com menos detalhes. (Freitas, 2020).

Figura 5 - Exemplo de diferentes resoluções de imagem



Fonte: Canaltech, 2020.

De acordo com Freitas (2020), a medida DPI é utilizada em imagens impressas, já a medida PPI é utilizada em imagens digitais, ponto de interesse deste estudo. Conforme visto na Figura 5, a quantidade de pixels interfere no quão legível é um caractere em uma imagem. Isso significa que a capacidade de resolução da câmera utilizada para captura da imagem no momento do escaneamento da lista de ingredientes irá interferir no resultado final da extração.

3 TRABALHOS RELACIONADOS

Esta seção apresenta softwares relacionados ao objeto deste estudo. A busca de trabalhos relacionados é uma etapa fundamental para o conhecimento do estado da arte dos sistemas disponíveis para o público. Por intermédio da verificação de trabalhos semelhantes é possível identificar as principais funcionalidades de sistemas que utilizam o reconhecimento óptico de caracteres para o escaneamento de ingredientes cosméticos. Isso possibilita apontar novos desafios. No caso desta dissertação, a função principal é a informação e descrição dos ingredientes cosméticos aos consumidores de produtos de beleza.

Com base nos objetivos deste estudo, foi identificado que o nome comum para esse tipo de software no mercado é “cosmetic scan”, em português “escaneamento de cosméticos”. Para tanto, foi realizada uma busca pelo termo "cosmetic scan" no buscador *Google*, excluindo resultados que não estão relacionados a cosméticos. Desta forma, foram selecionadas oito aplicações: *INCI Beauty*⁶, *CodeCheck*⁷, *Cosmetic Scan*⁸, *Cosmeticon*⁹, *CosmoSweet*¹⁰, *INCIdecoder*¹¹, *OnSkin*¹² e *Yuka*¹³ que fazem a leitura e identificação dos ingredientes, seja por via OCR ou por escaneamento do código de barras trazendo a descrição e a análise dos ingredientes cosméticos por base de produtos pré-cadastrados na ferramenta.

Os quesitos de comparação foram selecionados com base nas funcionalidades comuns encontradas nas aplicações. Essas foram classificadas como:

- III. Aquisição da lista de ingredientes: OCR, código de barras ou QR Code;
- IV. Busca por produtos e ingredientes: por nome do ingrediente, por nome do produto/marca, busca por produtos que contém o ingrediente;
- V. Classificação do produto: com base em métodos desenvolvidos pela equipe do software;
- VI. Recomendações de alternativas melhores ao cosmético escaneado;
- VII. Tipo de aplicação: web ou nativa (Android e/ou IOS);
- VIII. Histórico dos produtos escaneados.

⁶ <https://incibeauty.com/en>

⁷ <https://codecheck-app.com/>

⁸ <https://opencosmetics-project.web.app/>

⁹ <https://apps.apple.com/us/app/cosmetic-scanner-cosmeticon/id1523733980>

¹⁰ <https://cosmosweet.com/en/>

¹¹ <https://incidecoder.com/>

¹² <https://onskin.ai/>

¹³ <https://yuka.io/en/>

O Quadro 1 apresenta uma comparação entre os softwares encontrados e o software desenvolvido nesta dissertação. O módulo de escaneamento deste estudo, por ser uma aplicação baseada na Web, funciona tanto nos dispositivos iOS e Android, sem a necessidade de instalação da aplicação no dispositivo do usuário. Assim a aplicação não consome espaço de armazenamento, sendo um diferencial competitivo frente aos seus concorrentes.

Quadro 1 - Trabalhos relacionados

FUNCCIONALIDADES	APLICATIVO	INCI BEAUTY	CODE CHECK	COSMETIC SCAN (BEAUTY SCAN)	COSMETICON	COSMOSWEET	INCIDECODER	ONSKIN	YUIKA
OCR	x			x	x		x		
QR CODE			x						
CÓDIGO DE BARRAS	x	x	x			x		x	x
BUSCA DE INGREDIENTES	x					x	x		
BUSCA PELO NOME DO PRODUTO OU MARCA	x	x	x				x	x	x
BUSCA DE PRODUTOS POR INGREDIENTE						x	x		
CLASSIFICAÇÃO DOS INGREDIENTES E COSMÉTICO	x	x	x	x	x	x	x	x	x
RECOMENDAÇÃO DE ALTERNATIVAS	x	x	x					x	x
APLICAÇÃO WEB	x					x	x		
APLICAÇÃO MÓVEL NATIVA	x	x	x	x				x	x
HISTÓRICO	x	x	x	x	x	x	x	x	x

Fonte: Elaborado pela autora.

Dos softwares relacionados, apenas *CodeCheck*⁵ oferece suporte a leitura de QRCode. É importante ressaltar que para atender a resolução RDC N° 773, muitas fabricantes de cosméticos estão disponibilizando a lista dos ingredientes via QRCode. Por hora os ingredientes são listados com seu nome INCI outrora na língua portuguesa. Isto é um desafio para o armazenamento e performance na consulta dos dados, visto que o volume gerado de informações armazenadas pode ser elevado ao dobro, pois além da consulta pelo nome INCI também é necessário a busca pelo nome em português.

Para os sistemas que utilizam a leitura do código de barras para listar os ingredientes, é necessário que a lista dos ingredientes do produto alvo esteja salva em uma base de dados. Devido a grande quantidade de cosméticos disponíveis no mercado, a tendência de consumo de recursos de armazenamento tende aumentar. A facilidade para o usuário de escanear o código de barras é maior do que a captura da imagem da lista de ingredientes, o que exige uma certa habilidade manual, justificando a implementação futura dessa funcionalidade.

Outra categoria de sistemas encontrada relacionada ao escaneamento de cosméticos são aplicativos cuja função é indicar a data de fabricação, prazo de validade e vida útil após a abertura da embalagem. É o caso do *CheckFresh*¹⁴, *Cosmetic Checker*¹⁵, *Cosmetics Wizard*¹⁶ e *Check Cosmetic*¹⁷. Embora não façam a leitura, descrição e análise dos ingredientes cosméticos, os mesmos realizam a busca das informações pela leitura do código de barras e/ou indicação da marca e número do lote do produto.

Em tempo, mesmo que este estudo seja relacionado a ingredientes cosméticos, é importante destacar que estudos utilizando OCR para identificar substâncias em outras áreas do conhecimento estão sendo realizados como, por exemplo, a identificação de substâncias alérgicas em alimentos proposta pelo aplicativo *Yuka*¹¹. Outra função interessante encontrada no aplicativo *CodeCheck*⁵ é a indicação de preferências pelo usuário, onde podem ser selecionados produtos veganos, livre de glúten, micro-plásticos e lactose.

Para a funcionalidade de classificação dos ingredientes são atribuídas notas de acordo com os seus potenciais efeitos e impactos na saúde ou no meio ambiente. Com base em informações de artigos científicos e órgãos reguladores, cada aplicativo desenvolveu seu método de pontuação das fórmulas, avaliando se a mesma pode conter ingredientes com desregulador endócrino, formaldeídos, cancerígeno, alergênico, irritante ou poluente.

Podemos concluir que o desenvolvimento de aplicações que utilizam a tecnologia de reconhecimento óptico de caracteres ou escaneamento de código de barras vai muito além da descrição dos ingredientes da fórmula de um cosmético para o usuário. Estes softwares fornecem uma análise detalhada ao consumidor acerca da qualidade dos ingredientes, alguns oferecendo alternativas de produtos com melhores ingredientes.

¹⁴ <https://www.checkfresh.com/>

¹⁵ <https://cosmeticchecker.app/>

¹⁶ <https://cosmeticswizard.net/>

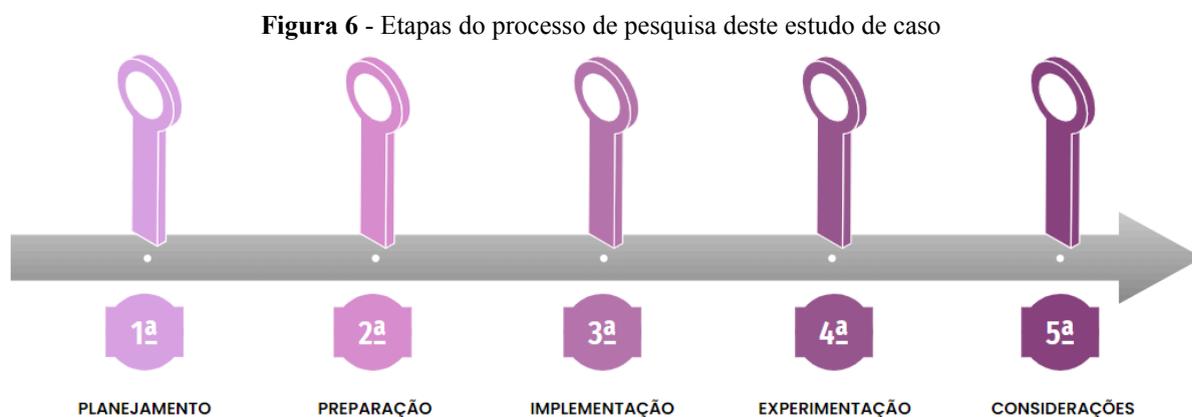
¹⁷ <https://checkcosmetic.net/>

4 MÉTODOS E PROCEDIMENTOS DA PESQUISA

Este capítulo descreve o delineamento da pesquisa pelos preceitos da engenharia de software experimental, uma subárea da engenharia de software que tem por objetivo melhorar métodos, processos e ferramentas para a engenharia de software. Este estudo primário desenvolve-se usando o método empírico empregando a estratégia de estudo de caso.

De acordo com Wohlin *et al.* (2012), o estudo de caso em engenharia de software experimental é conduzido para investigar uma única entidade ou fenômeno em seu contexto de vida real, em um espaço de tempo específico. O nível de controle é menor em um estudo de caso do que em um experimento. Um estudo de caso é um estudo observacional, enquanto o experimento é um estudo controlado, geralmente a cargo dos laboratórios. O fator que levou a escolha desta estratégia de pesquisa é de que estudos de caso têm a possibilidade de controle na medição dos resultados a um custo menor do que um experimento.

Em Wohlin *et al.* (2012), um estudo de caso pode ser dividido em cinco etapas: desenho do estudo, preparação para a coleta de dados, coleta dos dados, análise dos dados coletados e relatórios. Assim, o estudo dividiu-se em cinco etapas, conforme mostrado na Figura 6.



Fonte: Elaborado pela autora.

4.1 Planejamento

Na etapa de planejamento, identificou-se a oportunidade e os objetivos do estudo. Definido a estratégia de pesquisa, passou-se para a criação dos artefatos a serem desenvolvidos. Para manter uma boa coesão entre os componentes do sistema, foi planejado o estilo arquitetural modular do software. Cada módulo possui uma responsabilidade única dentro do domínio da aplicação. Este estilo de arquitetura permite uma fácil manutenção e evolução do software.

Também fez parte do planejamento, a definição dos requisitos funcionais e não funcionais do módulo. Os requisitos não funcionais descrevem a forma como o sistema opera e suas restrições. Para tanto, os requisitos não funcionais do módulo incluem o funcionamento da ferramenta em ambiente Web e o dataset em JSON.

Os requisitos funcionais descrevem os serviços realizados pelo sistema para atender a uma solicitação do usuário. São os requisitos funcionais do módulo a permissão do upload da imagem já armazenada no celular ou a captura instantânea da imagem através da câmera, a leitura da imagem enviada, o reconhecimento óptico dos caracteres da imagem, o formulário para consulta de ingredientes pelo nome INCI e descrição dos ingredientes.

4.2 Preparação

Conforme Gonçalves (2021), a definição da arquitetura de um software é o primeiro passo para a solução de um problema, antes de sua implementação. Esta determina como um software será construído e como irá se comportar. O sistema *Belezati* foi desenvolvido em Angular na versão 17. A escolha por Angular foi feita por ser um framework que possui suporte ao desenvolvimento modular e conter uma vasta lista de recursos próprios frente ao seu maior concorrente no mercado, o React. O Tesseract foi a primeira escolha de mecanismo de OCR para o desenvolvimento deste estudo. Ele é gratuito e de código aberto, muito conhecido na comunidade e contém um vasto volume de estudos relacionados.

Após os primeiros testes com o Tesseract.js, foi percebido uma maior necessidade de ajustes na qualidade e perspectiva das imagens para o reconhecimento dos caracteres. Neste momento, foram buscadas alternativas. O Textract da AWS foi a segunda escolha de mecanismo de OCR, por possuir recursos avançados para a extração dos caracteres utilizando *Machine Learning*.

A coleta dos dados foi realizada através da observação das composições dos produtos diretamente das embalagens, a fim de compreender a forma como comumente os ingredientes cosméticos são listados no mercado. Em seguida, através de pesquisas bibliográficas, foi criado o dataset, utilizando o formato de dados JSON com as descrições dos ingredientes para posterior cruzamento do texto extraído com o significado encontrado no dataset. Além da observação dos ingredientes nas embalagens e pesquisa em documentos bibliográficos, também fez parte da coleta de dados as informações fornecidas pela ANVISA, acerca da tradução, função dos ingredientes e forma aceita para listar os ingredientes nas embalagens dos produtos utilizando o nome INCI.

4.3 Implementação

O desenvolvimento do módulo de escaneamento de ingredientes foi realizado na IDE *Visual Studio Code* da Microsoft, usando uma arquitetura em camadas. O módulo contém uma camada de interação com o usuário e apresentação dos dados, uma camada de serviço que faz o acesso às APIs de OCR e busca dos ingredientes pelo nome INCI no dataset, e uma camada de entidade que é uma representação dos dados descrevendo os atributos do ingrediente.

4.4 Experimentos

Após a conclusão da implementação, os testes foram iniciados incluindo quatro tipos de imagens capturadas de um iPhone 13 com IOS 17.2. Foi feito também o cálculo de acurácia e registro do tempo de execução do algoritmo. Para analisar a acurácia do reconhecimento óptico dos caracteres foi executada uma função para realizar a relação de pertencimento entre dois conjuntos. O conjunto A define os caracteres extraídos via OCR e o conjunto B, o texto real escrito na embalagem do produto, conforme representado na Figura 7.

Figura 7 - Função de cálculo de Acurácia

$$g(A, B) = \begin{cases} 1 & \text{se } \exists x \in A, x \in B \\ 0 & \text{caso contrário} \end{cases}$$

Fonte: Elaborado pela autora.

A função retorna 1 caso um elemento do conjunto A também pertença ao conjunto B. Após, é feita a soma dos caracteres de B e dividido pela quantidade de caracteres correspondentes encontrados. Por fim, é feita a divisão por 100, para chegar ao valor da porcentagem de acurácia.

Para analisar a performance da extração, como demonstrado na Figura 8, foi capturado o tempo antes de chamar a função de extração com o método “*now*” do objeto “*Performance*” do Node.js e armazenado na constante “início”. Após a chamada da função de extração, foi feita a captura do tempo após a execução armazenando na constante “fim” também utilizando o objeto “*Performance*”. Subtraindo o tempo final pelo tempo inicial, dividindo por 1000 para transformar os milissegundos em segundos, chega-se ao tempo de execução.

Figura 8 - Código do cálculo de performance

```
const inicio = performance.now();
this.tesseractService.extraction(this.imgUrl).then(dados => {
  console.log("TESSERACT:" + dados);
  const fim = performance.now();
  const diferenca = (fim - inicio) / 1000;
  console.log(`Tempo de execução: ${diferenca.toFixed(2)} segundos`);
});
```

Fonte: Elaborado pela autora.

4.5 Limitações do Estudo

Este estudo se limita ao desenvolvimento de um módulo dentro do sistema *Belezati*. Ele foi desenvolvido em Angular, não fazendo parte deste estudo a implementação completa da solução do sistema e o desenvolvimento da autenticação e autorização do uso do módulo.

Outra limitação encontrada é referente ao posicionamento da fotografia enviada e o tratamento das imagens na fase anterior a segmentação do processo de reconhecimento de caracteres. Visto que, durante os estudos, a ferramenta Textract já realiza esse melhoramento internamente de forma muito efetiva, não sendo de interesse da pesquisa a realização dessa atividade, considerando que um dos objetivos do estudo é testar a performance das APIs públicas de OCR.

Além dessas limitações, o módulo desenvolvido não faz a busca por palavras similares. Caso o reconhecimento óptico da palavra tenha gerado um caractere divergente da embalagem, como o caso da troca de caracteres “Y” por “V”, ou de “O” por “0”, “;” por “:” e “A” por “4”, nenhum tratamento é feito.

Durante os experimentos também foi constatado que para analisar embalagens cilíndricas, seria necessário acesso a funcionalidade de captura de imagem de forma panorâmica. Uma das alternativas poderia ser a criação da aplicação de forma nativa nos aparelhos de IOS e Android, mais comuns no mercado, para ter acesso total às funcionalidades da câmera. Embora esta funcionalidade estivesse disponível para Web, ainda a prática do uso no dia a dia não a torna viável, visto que capturar imagens de forma panorâmica necessita uma maior habilidade manual do usuário, e a aplicação foi desenhada para ser usada da forma mais simples possível. Foi também pensando na possibilidade de captura dos caracteres por vídeo, mas as ferramentas testadas não oferecem suporte para o reconhecimento óptico de caracteres em arquivos de vídeo.

5 ARQUITETURA DO MÓDULO

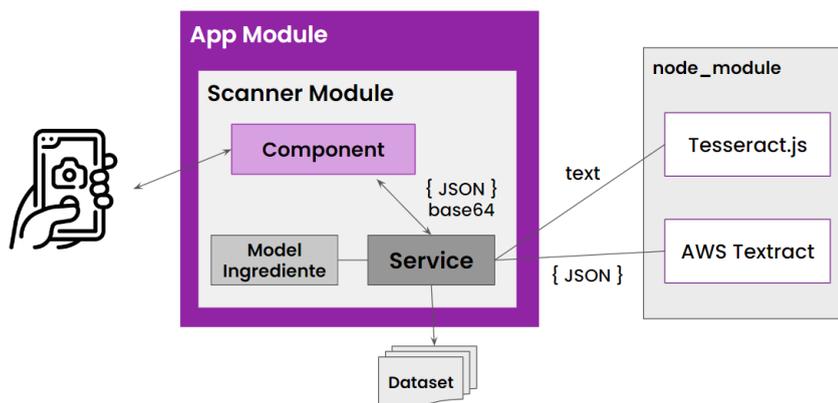
Este capítulo apresenta a arquitetura de software implementada no módulo de escaneamento de ingredientes do sistema *Belezati*. O nome do sistema foi definido pela junção da palavra “Beleza”, assunto tema deste estudo; e a sigla “TI”, de Tecnologia da Informação, área de concentração desta pesquisa.

5.1 Módulos do sistema

O sistema foi estruturado em módulos. Cada módulo é responsável por um domínio de negócio da aplicação, conforme Figura 9. Cada módulo foi dividido em três camadas:

- Componente: Responsável por apresentar as informações aos usuários e requisitar dados para a camada de serviços.
- Serviço: responsável por conectar aos serviços de OCR e de busca no dataset;
- Entidade: responsável por representar a entidade.

Figura 9 - Estrutura do módulo de escaneamento de ingredientes cosméticos



Fonte: Elaborado pela autora.

O Angular cria um arquivo chamado `package.json`, onde listam-se todas as dependências necessárias para a execução do projeto. Ao rodar pela primeira vez o programa, este arquivo é invocado e o *download* das dependências na pasta `node_module` é realizado. Para este módulo, possui-se a dependência do módulo do Tesseract.js, que faz acesso ao Tesseract e o AWS SDK, que contém o cliente de acesso à ferramenta Amazon Textract.

Cada componente contém arquivos HTML, CSS e TypeScript necessários para a visualização, estilização e comportamento da tela. O módulo aceita o envio de imagens nos formatos `.jpg` `.jpeg` `.png`. Para que a foto capturada seja reconhecida, seu conteúdo é lido com o objeto `FileReader`, invocando o método `readAsDataURL()`, onde ele retorna os dados da imagem contendo o formato da imagem e seu conteúdo codificado em base64.

Para fazer o reconhecimento dos caracteres via Tesseract, o envio do arquivo em uma string de base64 é o suficiente. Já para o Amazon Textract é necessário decodificar os dados da imagem que estão em base64 para bytes, onde cada caractere em byte irá ser trocado por um número correspondente em Unicode e armazenado em um array. Após encontrar as correspondências, o array gerado é convertido em um array de 8 bits de inteiro e enviado para o Amazon Textract fazer o reconhecimento dos caracteres. Após o reconhecimento pelas ferramentas de OCR, é uma string é recebida como retorno e armazenado em uma variável, para então ser exibido ao usuário. Posteriormente é invocado o método de tradução, onde é feito a busca das palavras do dataset dentro do texto extraído.

5.2 Esquema do dataset

Um esquema, palavra que vem do termo em latim *schema*, descreve ou representa coisas materiais ou abstratas. Neste sentido, o esquema do dataset de ingredientes cosméticos está representado com as seguintes propriedades:

- INCI: nome INCI do ingrediente usado como índice do elemento;
- nome: nome em português seguindo a tradução da ANVISA;
- tipo: informa se o elemento é um ingrediente ou substância;
- funcao: descreve a função do elemento de acordo com a ANVISA, conforme anexo A;
- descricao: descreve as características do elemento.

Neste esquema, o nome INCI é utilizado como chave indexadora para localizar as propriedades do ingrediente. Como regra, ele deve ser escrito em caixa alta.

A escolha por criar a coleção de ingredientes no formato de arquivo JSON vai da possibilidade do usuário poder consultar as informações sobre os ingredientes, mesmo sem acesso a internet, digitando o nome dos ingredientes no formulário, pois o download do dataset no aparelho do usuário é feito no primeiro acesso do usuário ao sistema. O arquivo JSON fica disponível em cache e só é atualizado caso seja feita uma nova requisição ao servidor. Para o processo de reconhecimento dos caracteres via OCR ainda é necessário uma conexão com a internet para acessar as APIs externas ao módulo.

Desta forma, o usuário pode utilizar os recursos do sistema que estão disponíveis em cache mesmo em ambientes onde o sinal da internet é ruim ou inexistente. Como o número de ingredientes atuais no mercado gira em torno de 12 mil, e a criação de novos ingredientes não é algo que ocorra a todo momento na indústria química devido ao fato de que para disponibilizar um ingrediente são necessários testes e autorizações de órgãos reguladores, o tamanho do arquivo tende a se manter estável mesmo em um grande período de tempo.

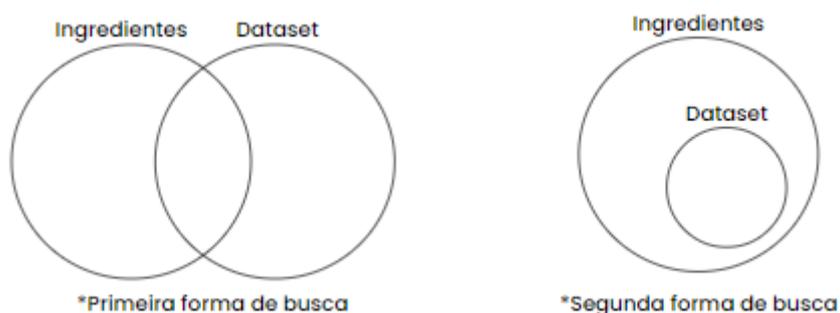
6 EXPERIMENTOS E ANÁLISES

Para avaliar a acurácia e performance dos dois mecanismos de OCR foram selecionadas quatro tipos de embalagens encontradas no mercado: embalagem clara e caracteres escuros, embalagem escura com caracteres claros, embalagem colorida com caracteres coloridos e embalagem com baixo contraste.

Na primeira tentativa de busca dos ingredientes, o sistema buscava todas as palavras depois do termo “ingredientes:”, e a cada vírgula encontrada, o sistema entendia que se tratava do nome de um ingrediente, adicionando-o em um array para posterior consulta no dataset. O problema nesta forma de busca era que, em muitas embalagens, o início da lista de ingredientes poderia vir com o termo “composição”, “*ingredients*” ou “fórmula”, ou ainda os ingredientes poderiam vir separados por ponto e vírgula. Ainda poderia haver a perda de ingredientes caso o OCR omitisse a vírgula por baixa resolução da imagem.

Para resolver este problema, o módulo de escaneamento passou a ler todo o texto contido no rótulo da embalagem do produto. Assim, nenhuma palavra capturada seria perdida. E ao invés de realizar a intersecção da lista de ingredientes extraída com as palavras no dataset, como demonstrado na Figura 10, a busca passou a ser feita pelos termos do dataset contidos dentro do conjunto de todas as palavras extraídas pelo OCR. Desta forma, mesmo em ingredientes onde o nome é escrito em duas palavras, como o caso de “Propylene Glycol”, que anteriormente poderia ser perdido caso a vírgula fosse omitida no momento da extração, agora podem ser encontrados facilmente. O que poderia ser um desafio, como a lista de Ingredientes em duas colunas, ou com a nova resolução da ANVISA de inclusão dos ingredientes em português, onde muitas empresas optaram por colocar o nome em português ao lado do nome INCI, como no caso da embalagem de shampoo para bebês, foi resolvido com este método de busca.

Figura 10 - Exemplo de busca pelos ingredientes no dataset



Fonte: Elaborado pela autora.

Para que não haja perda de informação no momento da busca dos ingredientes no dataset, o texto extraído foi então convertido para letras maiúsculas antes do cruzamento dos dados, conforme o registrado no arquivo JSON, com a chave que utiliza o nome INCI como indexador. Assim um ingrediente só não seria localizado no dataset caso sua escrita esteja errada como por exemplo ao invés de ser “AQUA” estar “AQU4”.

6.1 Embalagem clara e caracteres escuros

A primeira embalagem capturada é de um shampoo em frasco branco e caracteres pretos, facilmente encontrado em supermercados. Os caracteres impressos no rótulo da Figura 11 são demasiadamente pequenos, sendo de difícil leitura até mesmo a olho nu. O Quadro 2 analisa comparativamente Tesseract.js e Amazon Textract em relação a esta embalagem.

Figura 11 - Shampoo em embalagem branca



Fonte: Capturado pela autora.

Lista de ingredientes na embalagem: AQUA, SODIUM LAURETH SULFATE, COCAMIDOPROPYL BETAINE, SODIUM CITRATE, SODIUM XYLENESULFONATE, GLYCOL DISTEARATE, SODIUM CHLORIDE, DIMETHICONE, PARFUM, STEARYL ALCOHOL, CITRIC ACID, SODIUM BENZOATE, POLYQUATERNIUM-6, CETYL ALCOHOL, TETRASODIUM EDTA, HEXYL CINNAMAL, PANTHENOL, PANTHENYL ETHYL ETHER, HISTIDINE, TOCOPHERYL ACETATE, PRUNUS AMYGDALUS DULCIS OIL, PEG-40 HYDROGENATED CASTOR OIL, GLYCERIN, COCOS NUCIFERA OIL, ARGANIA SPINOSA KERNEL OIL, ALOE BARBADENSIS LEAF EXTRACT, HYDROLYZED WHEAT PROTEIN, RUBUS IDAEUS FRUIT EXTRACT, HYDROLYZED SILK, METHYLCHLOROISO THIAZOLINONE HYDROLYZED WHEAT STARCH, METHYLISOTHIAZOLINONE.

Quadro 2 - Análise embalagem clara e caracteres pretos

Tesseract.js	Amazon Textract
CHONDE, DIMETHCONE, PARFUM, STEARYL ALCOHOL TRC ACD, Sop I, (E ALCOHOL TETRASODIUM EDTA, HEXYL CINNAMAL PEND, WITONE, TOCOPHERVL ACETATE, RUNS AMYGDALLS DIL OU, PEGAO LG COCOS NUCIFERA ON, ARGANIA SPINOSA KERNEL OI ALOE DROLYZED WINEAT PROTEIN, RUBUS IDAEUS FRUIT EXTAC, ORG) 3 OUNORE, HYOROLZED WHEAT STARCK, METEVUSOTHIAZOLHONE	AQUA, SODIUM LAURETH SULFATE COCAMIDOPROPYL BETAINE, SODIUM CITRATE. SODIUM XYLENESULFONATE GLYCO DISTEARATE SODIUM CHLORIDE DIMETHICONE, PARFUM, STEARYL ALCOHOL CITRIC ACID, SODIUM BENZOATE, POLYQUATERNIUM-6. CETYL ALCOHOL TETRASODIUM EDTA HEXYL CINNAMAL PANTHENOL PANTHENYL ETHYL ETHER, HISTIDINE, TOCOPHERYL ACETATE, PRUNUS AMYGDALUS DULCIS OIL PEG-40 HYDROGENATED CASTOR OIL GLYCERIN. COCOS NUCIFERA OIL ARGANIA SPINOSA KERNEL OIL, ALOE BARBADENSIS LEAF EXTRACT HYDROLYZED WHEAT PROTEIN RUBUS IDAEUS FRUIT EXTRACT HYDROLYZED SILK METHYLCHLOROISOTHIAZOLINONE HYDROLYZED WHEAT STARCH, METHYLISOTHIAZOLINONE
A acurácia é de: 2.08% Tempo de execução: 13.17 segundos	A acurácia é de: 8.79% Tempo de execução: 3.62 segundos

Fonte: elaborado pela autora.

6.2 Embalagem escura e caracteres claros

A segunda embalagem, também é de um shampoo, encontrado em uma loja de cosméticos. Na embalagem, os caracteres estão maiores, conforme a Figura 12, e o reconhecimento da pontuação como o ponto e vírgula foi facilmente identificado pelas ferramentas de OCR, resultando em uma acurácia maior. O Quadro 3 analisa Tesseract.js e Amazon Textract em relação à embalagem escura e caracteres claros.

Figura 12 - Shampoo em embalagem preta



Fonte: Capturado pela autora.

Lista de ingredientes na embalagem: AQUA (WATER); SODIUM LAURETH SULFATE; COCAMIDOPROPYL BETAINE; PARFUM (FRAGRANCE); POTASSIUM CHLORIDE; COCO-GLUCOSIDE; DIMETHICONOL; GLYCERYL OLEATE; PHENOXYETHANOL; SODIUM BENZOATE; GLYCOL DISTEARATE; GUAR HYDROXYPROPYLTRIMONIUM CHLORIDE; LAURETH-4; ACRYLATES/C10-30 ALKYL ACRYLATE CROSSPOLYMER; LACTIC ACID; SODIUM GLUCONATE; GLYCERIN; SODIUM HYDROXIDE; CITRIC ACID, TEA-DODECYLBENZENESULFONATE; BENZOIC ACID; POLYSORBATE 20; TRITICUM VULGARE PROTEIN [TRITICUM VULGARE (WHEAT) PROTEIN]; BENZYL ALCOHOL; TOCOPHEROL; CAPRYLYL GLYCOL; 1,2-HEXANEDIOL; HYDROGENATED PALM GLYCERIDES CITRATE; CAULERPA LENTILLIFERA EXTRACT2, HYDROLYZED QUINOA; ZEA MAYS STARCH [ZEA MAYS (CORN) STARCH]; IODOPROPYNYL BUTYLCARBAMATE; Ingredientes POTASSIUM SORBATE; GLUCONOLACTONE; PHENETHYL ALCOHOL; TOCOPHEROL (português) BENZYL SALICYLATE; CITRONELLOL; COUMARIN; LIMONENE; LINALOOL.

Quadro 3 - Análise embalagem escura e caracteres brancos

Tesseract.js	Amazon Textract
<p>SIEATE, COCAMDOPROPYL BETAINE; PARFUM (FRAGRANÇE); POTASSIUM CHLORDE; COCO-GLCORDE; DIMETHCONOL; GLYCERYL OLEATE; PHENOXYETHANOL; SODIUM BENZOATE; GLYCOL DISTEARTE; GUI HIDEOPROPTLTRMONUM CHLORIDE; LAURETH-A; ACRYLATES/CIO-S0 ALKYL ACRIUATE CROSSOLVER CTC ACD, SODUM GLUCONATE, GLYCERIN; SODIUM. HIDRODE, OTC _ A: 4 TEADODECYLEENTENESULFONATE; BENZOIC ACD; POLYSORBATE 20; TRIICUN VULGARE PROTEN TT AGE MEN) FROTENE BEL ALCOHOL: TOCOPHEROL CFU GUC, LAA: HYDROGENATED PALM GLYCERIDES CITRATE; CAULERPA LENTILUFERA EXTRACT: HIDROUZED QUBOK HIS STARCH [EA MAIS (CORN) STARCH, IODOPROPINL BUTMLCARBAÁTE er, PCTS SORBATE: GucwucTe: REETHL ACH ORR oud EO SALCYLATE CTRONELLOL COUMARIN; LIMONENE; LNALOOOL</p>	<p>AQUA (WATER): SODIUM LAURETH SULFATE; COCAMIDOPROPYL BETAINE: PARFUM (FRAGRANCE): POTASSIUM CHLORIDE: COCO-GLUCOSIDE: DIMETHICONOL- GLYCERYL OLEATE: PHENOXYETHANOL; SODIUM BENZOATE: GLYCOL DISTEARATE: GUAR HYDROXYPROPYLTRIMONIUM CHLORIDE: LAURETH-4; ACRYLATES/C10-30 ALKYL ACRYLATE CROSSPOLYMER LACTIC ACID: SODIUM GLUCONATE: GLYCERIN: SODIUM HYDROXIDE: CITRIC ACID, A-CODECYLBENZENESULFONATE BENZOIC ACID: POLYSORBATE 20: TRITICUM VULGARE PROTEIN ITRITICUM VULGARE (WHEAT) PROTEIN]; BENZYL ALCOHOL: TOCOPHEROL CAPRYLYL GLYCOL: 1,2-HEXANEDICE HYDROGENATED PALM GLYCERIDES CITRATE: CAULERPA LENTILLIFERA EXTRACT HYDROLYZED QUINOA; ZEA MAYS STARCH ZEA MAYS (CORN) STARCH]; IODOPROPYNYL BUTYLCARBAMATE Ingredientes POTASSIUM SORBATE: GLUCONOLACTONE: PHENETHYL ALCOHOL: TOCOPHEROL (portugues) BENZYL SALICYLATE: CITRONELLOL COUMARIN; LIMONENE; LINALOOL</p>
<p>A acurácia é de: 5.24% Tempo de execução: 45.62 segundos</p>	<p>A acurácia é de: 36.09% Tempo de execução: 4.25 segundos</p>

Fonte: elaborado pela autora.

A acurácia neste experimento foi maior pois além dos caracteres na embalagem estarem maiores que a embalagem anterior, o espaçamento entre eles também estavam, sendo mais legível até mesmo a olho nu. Pode-se inferir neste momento que o que determina a

acurácia não é a cor do caractere e sim o quão legível ele está na embalagem, como o tamanho da fonte maior e o espaçamento entre os caracteres adequados.

6.3 Embalagem colorida e caracteres coloridos

Na embalagem de um shampoo para bebês da Figura 13, encontrado em supermercados e farmácias, o rótulo encontra-se em três cores, branco, amarelo e azul, e em formato de gota. As informações de interesse encontram-se em laranja. O Quadro 4 compara Tesseract.js e Amazon Textract em relação à embalagem e caracteres coloridos..

Figura 13 - Shampoo para bebê



Fonte: Capturado pela autora.

Lista de ingredientes na embalagem: Aqua/água purificada, Cocamidopropyl Betaine/cocoamidopropilbetaína, Decyl Glucoside/decil glicosídeo, Sodium Cocoyl Isethionate/cocoil isetionato de sódio, PEG-80 Sorbitan Laurate/polissorbitato 20, PEG-150 Distearate/diestearato de PEG-150, Glycerin/glicerol, Citric Acid/ácido cítrico, Sodium Benzoate/benzoato de sódio, Sodium Methyl Cocoyl Taurate/metil cocoil taurato de sódio, Parfum/fragrância, Disodium EDTA/edetato dissódico, Polyquaternium-10/poliquatérnio-10.

Quadro 4 - Análise embalagem colorida e caracteres coloridos

Tesseract.js	Amazon Textract
Não foi capaz de trazer os ingredientes INCI.	Cocoyl Isethionate/cocoil isetionato de sódio, PEG-80 Sorbitan Betaine/cocaamidopropileina, Decyl Glucoside/decil glicosídeo, Sodium INGREDIENTES: Aqua/água purificada, Cocamidopropy} citrico, 20. PEG-150 Sodium Distearate/diestearato Benzoate/benzoato Glycerin/glicerol, Laurate/polissorbato Cocoyl Citric Acid/ácido de PEG-150, cocoil de sódio, Sodium Methyl Taurate/meti taurato de sódio, Parfum/fragrancia, Disodium EDTA/edetato dissódico,
A acurácia é de: 0% Tempo de execução: 6.22 segundos	A acurácia é de: 4.26% Tempo de execução: 3.83 segundos

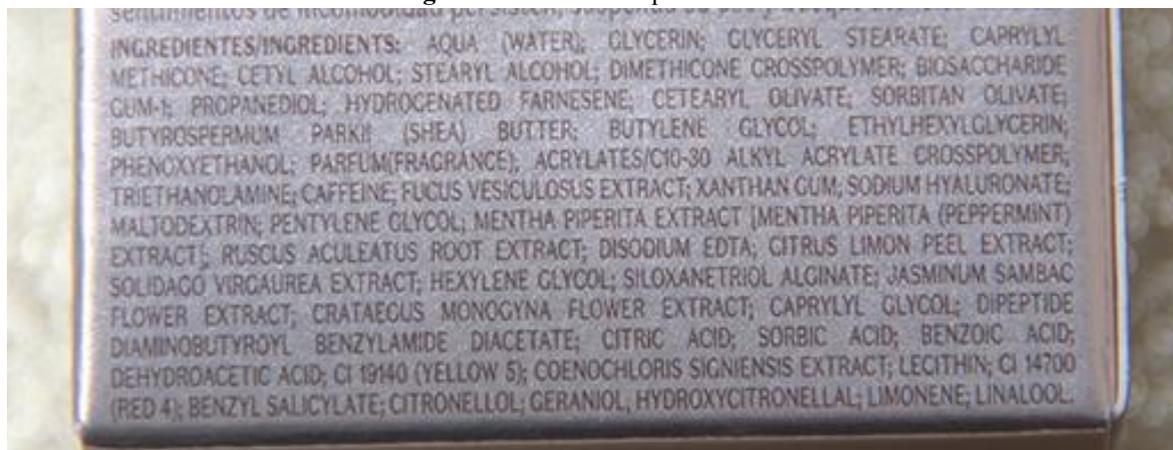
Fonte: Elaborado pela autora.

Neste experimento, o Tesseract não foi capaz de reconhecer os caracteres. Já o Amazon Textract não conseguiu posicionar as linhas corretamente dentro do parágrafo, colocando a palavra “ingredientes”, que consta no início da lista dos ingredientes, no meio do texto e trocando a ordem de alguns ingredientes. Isto não impede o sistema de encontrar os ingredientes no dataset, visto que ele não procura mais o nome dos ingredientes depois da palavra “ingrediente”, como exemplificado anteriormente.

6.4 Embalagem com baixo contraste e resolução

Neste experimento foi realizado um teste com a imagem em baixa resolução e os caracteres com cor muito próxima ao do fundo da embalagem, sinalizando um baixo contraste e nitidez, conforme Figura 14. O Quadro 5 analisa comparativamente Tesseract.js e Amazon Textract em relação à embalagem com baixo contraste e resolução.

Figura 14 - Hidratante para os olhos



Fonte: Coisas de Diva¹⁸, 2023.

¹⁸ <https://www.coisasdediva.com.br/2023/05/hialu-cafeina/>

Lista de ingredientes na embalagem: AQUA (WATER); GLYCERIN; GLYCERYL STEARATE; CAPRYLYL METHICONE; CETYL ALCOHOL; STEARYL ALCOHOL; DIMETHICONE CROSSPOLYMER; BIOSACCHARIDE GUM-1; PROPANEDIOL; HYDROGENATED FARNESENE; CETEARYL OLIVATE; SORBITAN OLIVATE; BUTYROSPERMUM PARKII (SHEA) BUTTER; BUTYLENE GLYCOL; ETHYLHEXYLGLYCERIN; PHENOXYETHANOL; PARFUM(FRAGRANCE); ACRYLATES/C10-30 ALKYL ACRYLATE CROSSPOLYMER; TRIETHANOLAMINE; CAFFEINE; FUCUS VESICULOSUS EXTRACT; XANTHAN GUM; SODIUM HYALURONATE; MALTODEXTRIN; PENTYLENE GLYCOL; MENTHA PIPERITA EXTRACT [MENTHA PIPERITA (PEPPERMINT) EXTRACT]; RUSCUS ACULEATUS ROOT EXTRACT; DISODIUM EDTA; CITRUS LIMON PEEL EXTRACT; SOLIDAGO VIRGAUREA EXTRACT; HEXYLENE GLYCOL; SILOXANETRIOL ALGINATE; JASMINUM SAMBAC FLOWER EXTRACT; CRATAEGUS MONOGYNA FLOWER EXTRACT; CAPRYLYL GLYCOL; DIPEPTIDE DIAMINO BUTYROYL BENZYLAMIDE DIACETATE; CITRIC ACID; SORBIC ACID; BENZOIC ACID; DEHYDROACETIC ACID; CI 19140 (YELLOW 5); COENOCHELOSIS SIGNIENSIS EXTRACT; LECITHIN; CI 14700 (RED 4); BENZYL SALICYLATE; CITRONELLOL; GERANIOL; HYDROXYCITRONELLAL; LIMONENE; LINALOOL.

Quadro 5 - Análise embalagem com baixo contraste e resolução

Tesseract.js	Amazon Textract
<p>DATO a aaa ES Sa SALAME A o ABSTRACT cf EDT INGREDIENTES/INGREDIENTS : AQUA - (WATER. "GLYCERIN: GUYCERYL STEARATE: CAPRY No» ta. st. cow: TICE CROSSPOLYER; BOSCO Gy OPED IODROGENÁTED ARSE. C er, seh ond wo E MAO SE I RN bas Erm kha CU spun mow Rea SLICE PAT GOS NEXT PETA STRAT NENT Pe (EEN a E</p>	<p>- - INGREDIENTES/INGREDIENTS: AQUA (WATER); GLYCERIN; GLYCERYL STEARATE; CAPRYLYL METHICONE; CETYL ALCOHOL: STEARYU ALCOHOL; DIMETHICONE CROSSPOLYMER: BIOSACCHARIDE GUM-1; PROPANEDIOL HYDROCENATED FARNESENE CETEARYL OLIVATE; SORBITAN OLIVATE: BUTYROSPERMUM PARKI (SHEA) BUTTER; BUTYLENE GLYCOL: ETHYLHEXYLOLYCERIN; PHENOXYETHANOL PAREUM(FRAGRANCE), ACRYLATES/C10-30 ALKYL ACRYLATE CROSSPOLYMER TRIETHANOLAMINE CAFFEINE; FUCUS VESICULOSUS EXTRACT; XANTHAN GUM; SODIUM HYALURONATE; MALTODEXTRIN; PENTYLENE GLYCOL; MENTHA PIPERITA EXTRACT [MENTHA PIPERITA (PEPPERMINT) EXTRACT RUSCUS ACULEATUS ROOT EXTRACT: DISODIUM EDTA; CITRUS LIMON PEEL EXTRACT; SOLIDACO VIRCAUREA EXTRACT; HEXYLENE GLYCOL; SILOXANETRIOL ALGINATE; JASMINUM SAMBAC FLOWER EXTRACT; CRATAECUS MONOGYNA FLOWER EXTRACT; CAPRYLYL GLYCOL; DIPEPTIDE DIAMINO BUTYROYL BENZYLAMIDE DIACETATE CITRIC ACID; SORBIC ACID; BENZOIC ACID; DEHYDROACETIC ACID; CI 19140 (YELLOW 5); COENOCHELOSIS SIGNIENSIS EXTRACT; LECITHIN; CI 14700 (RED 4); BENZYL SALICYLATE; CITRONELLOL GERANIOL, HYDROXYCITRONELLAL; LIMONENE; LINALOOL</p>
<p>A acurácia é de: 1.71% Tempo de execução: 9.41 segundos</p>	<p>A acurácia é de: 5.33% Tempo de execução: 3.07 segundos</p>

Fonte: Elaborado pela autora.

6.5 Embalagem posicionada em ângulos diversos

Para testar a capacidade de extração das ferramentas de OCR sem os caracteres posicionados em 0°, foram capturadas imagens com a embalagem lateralizada para esquerda, direita, 90° e em 180° conforme Figura 15, Figura 16, Figura 17 e Figura 18. O Quadro 6, Quadro 7, Quadro 8 e Quadro 9 analisam comparativamente Tesseract.js e Amazon Textract em relação a cada uma dessas inclinações.

Lista de ingredientes: AQUA (WATER); CAPRYLIC/CAPRIC TRIGLYCERIDE; DIMETHICONE; ALLANTOIN; NIACINAMIDE; PHENOXYETHANOL; CAPRYLYL GLYCOL; CAPRYLIC/CAPRIC GLYCERIDES; RETINOL; PARFUM (FRAGRANCE); TOCOPHEROL; POLYACRYLATE CROSSPOLYMER-6; DIMETHICONOL; POLYSORBATE 80; POLYSORBATE 20; OLEIC ACID; LINOLEIC ACID; XANTHAN GUM; PPG-15 STEARYL ETHER; STEARETH-2; STEARETH-21; BISABOLOL; MALTODEXTRIN; HYDROXYETHYL ACRYLATE/SODIUM ACRYLOYLDIMETHYL TAURATE COPOLYMER; PROPYLENE GLYCOL; ISOHEXADECANE; DISODIUM EDTA; POLYSORBATE 60; SORBITAN ISOSTEARATE; TRIETHANOLAMINE; LACTIC ACID; CI 19140 (YELLOW 5); VITIS VINIFERA SEED EXTRACT [VITIS VINIFERA (GRAPE) SEED EXTRACT]; COENOCHLORIS SIGNIENSIS EXTRACT; POTASSIUM SORBATE; LECITHIN; SODIUM SULFITE; TRISODIUM EDTA; CITRIC ACID; ALPHA-ISOMETHYL IONONE; CITRONELLOL; GERANIOL; HYDROXYCITRONELLAL; LIMONENE; LINALOOL.

Figura 15 - Embalagem virada para esquerda



Fonte: Capturado pela autora.

Quadro 6 - Análise embalagem virada para esquerda

Tesseract.js	Amazon Textract
Não foi capaz de trazer os ingredientes INCI.	AQUA (WATER); CAPRYLIC/CAPRIC TRIGLYCERIDE; DIMETHICONE; ALLANTOIN; NIACINAMIDE; PHENOXYETHANOL; CAPRYLYL GLYCOL; CAPRYLIC/CAPRIC GLYCERIDES; RETINOL; PARFUM (FRAGRANCE); TOCOPHEROL; POLYACRYLATE CROSSPOLYMER-6; DIMETHICONOL; POLYSORBATE 80; POLYSORBATE 20; OLEIC ACID; LINOLEIC ACID; XANTHAN GUM; PPG-15 STEARYL ETHER; STEARETH-2; STEARETH-21; BISABOLOL; MALTODEXTRIN; HYDROXYETHYL ACRYLATE/SODIUM ACRYLOYLDIMETHYL TAURATE COPOLYMER; PROPYLENE GLYCOL; ISOHEXADECANE; DISODIUM EDTA; POLYSORBATE 60; SORBITAN ISOSTEARATE; TRIETHANOLAMINE; LACTIC ACID; CI 19140 (YELLOW 5); VITIS VINIFERA SEED EXTRACT [VITIS VINIFERA (GRAPE) SEED EXTRACT]; COENOCHLORIS SIGNIENSIS EXTRACT; POTASSIUM SORBATE; LECITHIN; SODIUM SULFITE; TRISODIUM EDTA; CITRIC ACID; ALPHA-ISOMETHYL IONONE; CITRONELLOL; GERANIOL; HYDROXYCITRONELLAL; LIMONENE; LINALOOL.
A acurácia é de: 0% Tempo de execução: 37.15 segundos	A acurácia é de: 100% Tempo de execução: 3.07 segundos

Fonte: Elaborado pela autora.

Figura 16 - Embalagem virada para direita



Fonte: Capturado pela autora.

Quadro 7 - Análise embalagem virada para direita

Tesseract.js	Amazon Textract
Não foi capaz de trazer os ingredientes INCI.	AQUA (WATER); CAPRYLIC/CAPRIC TRIGLYCERIDE; DIMETHICONE; ALLANTOIN; NIACINAMIDE; PHENOXYETHANOL; CAPRYLYL GLYCOL; CAPRYLIC/CAPRIC GLYCERIDES; RETINOL; PARFUM DIMETHICONOL POLYSORBATE 80; POLYSORBATE 20; OLEIC ACID; LINOLEIC (FRAGRANCE); TOCOPHEROL; POLYACRYLATE CROSSPOLYMER-6 BISABOLOL; MALTODEXTRIN; HYDROXYETHYL ACRYLATE/SODIUM ACID; XANTHAN GUM; PPG-15 STEARYL ETHER; STEARETH-2; STEARETH-21; ACRYLOYLDIMETHYL TAURATE COPOLYMER; PROPYLENE GLYCOL; ISOSTEARATE; TRIETHANOLAMINE; LACTIC ACID; C 19140 (YELLOW 5); VITIS ISOHEXADECANE; DISODIUM EDTA; POLYSORBATE 60; SORBITAN COENOCHLORIS SODIUM SIGNIENSIS EXTRACT; POTASSIUM SORBATE; LECITHIN; VINIFERA SEED EXTRACT [VITIS VINIFERA (GRAPE) SEED EXTRACT CITRONELLOL CERANIOL; HYDROXYCITRONELLAL LIMONENE; LINALOOL SULFITE; TRISODIUM CITRIC ALPHA-ISOMETHYL IONONE;
A acurácia é de: 0% Tempo de execução: 13.72 segundos	A acurácia é de: 22.60% Tempo de execução: 4.72 segundos

Fonte: Elaborado pela autora.

Figura 17 - Embalagem posicionada em 90°



Fonte: Capturado pela autora.

Quadro 8 - Análise embalagem posicionada em 90°

Tesseract.js	Amazon Textract
Não foi capaz de trazer os ingredientes INCI.	AQUA (WATER); CAPRYLIC/CAPRIC TRIGLYCERIDE; DIMETHICONE; ALLANTOIN; NIACINAMIDE PHENOXYETHANOL CAPRYLYL GLYCOL; CAPRYLIC/CAPRIC GLYCERIDES; RETINOL; PARFUM (FRAGRANCE); TOCOPHEROL POLYACRYLATE CROSSPOLYMER-6; DIMETHICONOL POLYSORBATE 80; POLYSORBATE 20; OLEIC ACID; LINOLEIC ACID; XANTHAN GUM; PPG-15 STEARYL ETHER; STEARETH-2; STEARETH-21, BISABOLOL, MALTODEXTRIN; HYDROXYETHYL ACRYLATE/SODIUM ACRYLOYLDIMETHYL TAURATE COPOLYMER; PROPYLENE GLYCOL; ISOHEXADECANE DISODIUM EDTA; POLYSORBATE 60g SORBITAN ISOSTEARATE; TRIETHANOLAMINE; LACTIC ACID; CI 19140 (YELLOW 5); VITIS VINIFERA SEED EXTRACT [VITIS VINIFERA (GRAPE) SEED EXTRACT COENOCHLORIS SIGNIENSIS EXTRACT; POTASSIUM SORBATE; LECITHIN; SODIUM SULFITE TRISODIUM EDTA; CITRIC ACID; ALPHA-ISOMETHYL IONONE; CITRONELLOL GERANIOL; HYDROXYCITRONELLAL; LIMONENE; LINALOOL
A acurácia é de: 0% Tempo de execução: 13.72 segundos	A acurácia é de: 11.42% Tempo de execução: 4.72 segundos

Fonte: Elaborado pela autora.

Figura 18 - Embalagem posicionada em 180°



Fonte: Capturado pela autora.

Quadro 9 - Análise embalagem posicionada em 180°

Tesseract.js	Amazon Textract
	AQUA (WATER); CAPRYLIC/CAPRIC TRIGLYCERIDE; DIMETHICONE; ALLANTOIN; NIACINAMIDE PHENOXYETHANOL CAPRYLYL GLYCOL; CAPRYLIC/CAPRIC GLYCERIDES; RETINOL; PARFUM (FRAGRANCE); TOCOPHEROL POLYACRYLATE CROSSPOLYMER-6; DIMETHICONOL POLYSORBATE 80; POLYSORBATE 20; OLEIC ACID; LINOLEIC ACID; XANTHAN GUM; PPG-15 STEARYL ETHER: STEARETH-2; STEARETH-21; BISABOOL; MALTODEXTRIN: HYDROXYETHYL ACRYLATE/SODIUM ACRYLOYLDIMETHYL TAURATE COPOLYMER; PROPYLENE GLYCOL; ISOHEXADECANE; DISODIUM EDTA; POLYSORBATE 60; SORBITAN ISOSTEARATE; TRIETHANOLAMINE; LACTIC ACID; CI 19140 (YELLOW 5); VITIS VINIFERA SEED EXTRACT [VITIS VINIFERA (GRAPE) SEED EXTRACT]; COENOCHLORIS SIGNIENSIS EXTRACT; POTASSIUM SORBATE; LECITHIN; SODIUM SULFITE; TRISODIUM EDTA; CITRIC ACID; ALPHA-ISOMETHYL IONONE; CITRONELLOL; GERANIOL; HYDROXYCITRONELLAL; LIMONENE; LINALOOL
A acurácia é de: 0% Tempo de execução: 29.40 segundos	A acurácia é de: 14.42% Tempo de execução: 5.51 segundos

Fonte: Elaborado pela autora.

6.4 Problemas encontrados

Durante os experimentos, foi possível notar a troca de caracteres em imagens de baixa resolução e tamanho da fonte menor. Os erros mais comuns foram listados no Quadro 10 abaixo. A omissão de acentuação também foi percebida pois são caracteres muito pequenos como no caso da palavra “fragrância”.

Quadro 10 - Caracteres comumente trocados

Caractere Original	Caractere extraído
O	0
A	4
Y	V
;	:
G	C

Fonte: Elaborado pela autora.

7 CONSIDERAÇÕES FINAIS

O objetivo desta dissertação foi desenvolver um módulo de escaneamento de ingredientes cosméticos diretamente do rótulo das embalagens dos produtos em um sistema já existente, chamado *Belezati*. O sistema está desenvolvido em plataforma para Web, de arquitetura modular. Isso permite a utilização do mesmo em qualquer dispositivo moderno com acesso a internet, para que o público consumidor de cosméticos possa fazer melhores escolhas para suas necessidades estéticas com base em informações acessíveis, de fácil compreensão e de acordo com as normas da ANVISA. Para atingir este objetivo, duas ferramentas fornecedoras de tecnologia de reconhecimento óptico de caracteres via API foram implementadas neste trabalho: uma muito conhecida na comunidade de engenharia de software, por ser de código aberto e gratuita, a biblioteca Tesseract, e outra desenvolvida por uma das maiores empresas de tecnologia da atualidade, a Amazon Textract.

Conforme o decorrer dos experimentos, independente da cor das embalagens, a resolução, o tamanho dos caracteres e o espaçamento maior entre eles, demonstraram ser fatores determinantes para o sucesso da extração. Por não fazer parte do escopo desta pesquisa, o tratamento da imagem de pré-processamento, a posição da embalagem em 0° é a ideal para a extração dos caracteres, embora o Amazon Textract demonstrou possuir habilidade para extração mesmo com a embalagem posicionada em ângulos diferentes.

A ferramenta Amazon Textract mostrou-se superior em todos os testes realizados, tanto no tempo de reconhecimento dos caracteres, como na acurácia dos caracteres extraídos, em razão da ferramenta possuir internamente o tratamento da imagem em pré-processamento e técnicas de *Machine Learning*, o que permite reconhecer caracteres mesmo com algum aspecto adverso. Isso é diferente da ferramenta Tesseract, que para ter uma boa acurácia na extração, necessita que a embalagem na imagem esteja com a nitidez e contraste ajustados, e posicionada em 0° antes do envio para o reconhecimento. Devido a este fato, pelo custo de processamento no tratamento interno das imagens, a ferramenta Amazon Textract foi a escolha ideal para que seja mantida no módulo, mesmo que o processo de reconhecimento não seja gratuito como no Tesseract. Ainda assim, o custo e a velocidade do retorno das informações justificam seu uso, trazendo um maior conforto para os usuários, que terão o resultado mais rápido. Isto é um aspecto importante, pois o ambiente em que o usuário estará utilizando a ferramenta para auxiliar na escolha de um cosmético, como por exemplo em uma farmácia, necessita de agilidade do retorno das informações.

O desafio maior das ferramentas que utilizam o reconhecimento óptico de caracteres não é o processo de extração dos caracteres nas imagens. Esta tecnologia está muito avançada nos dias de hoje, sendo utilizada em vários meios da sociedade. A questão de lidar com o tratamento das informações extraídas que podem ter os caracteres reconhecidos de forma errônea é o maior desafio da engenharia de software, onde pode-se utilizar algoritmos para a localização de palavras similares. Neste caso, é preciso observar a performance da busca em grandes bases de dados.

Durante os experimentos deste estudo, foi constatado que o sistema auxilia na leitura da lista de ingredientes, onde os caracteres se encontram muito pequenos para pessoas com baixa visão ou distúrbios visuais. Este aspecto é importante pois demonstrou que o uso do reconhecimento óptico de caracteres é viável e deve ser mais explorado, pois promove a inclusão e independência de pessoas com estas condições.

7.1 Trabalhos Futuros

Os estudos na identificação de ingredientes cosméticos advindos de imagens, seja pela utilização da tecnologia OCR ou pela leitura do código de barras com o cruzamento dos ingredientes em uma base de dados ou dataset estão bem fundamentados no mercado. O que carece ainda são trabalhos que possibilitem a leitura do rótulo frontal do cosmético, que contém o nome do produto com tipagem grafada e logomarca das empresas, que necessitam de um tratamento diferenciado para o pré-processamento, segmentação e extração das características para o reconhecimento do texto e posterior descrição e avaliação do cosmético.

Ademais, devido a limitações do escopo desta pesquisa, os aspectos de extração de caracteres em embalagens curvas não foram contempladas, possibilitando um estudo futuro para a inclusão desta funcionalidade no software. Além da inclusão de algoritmos de reconhecimento de palavras similares, quando a extração de uma palavra trazer um caractere divergente da palavra original, o que pode gerar a não localização do ingrediente no dataset, pois o mesmo necessita da grafia exata para encontrar a correspondência no dataset.

7.2 Premiações

Durante o mestrado, o projeto foi submetido duas vezes ao desafio modelo de negócios inovadores do Circuito Startup Pampa, tendo obtido os seguintes resultados:

- 3º lugar em 2022: BELEZATI - Software para qualificar o atendimento estético.
- 1º lugar em 2023: BELEZATI - Software para qualificar o atendimento estético.

REFERÊNCIAS

- ANVISA. **Nomenclatura de ingredientes**. 06 ago. 2021. Disponível em: <<https://www.gov.br/anvisa/pt-br/acesoainformacao/perguntasfrequentes/cosmeticos/nomenclatura-de-ingredientes>>. Acesso em: 14 dez. 2023.
- ANVISA. **Tradução INCI**. 04 set. 2023. Disponível em: <<https://www.gov.br/anvisa/pt-br/assuntos/cosmeticos/traducao>>. Acesso em: 14 dez. 2023.
- BARELLI, Felipe. **Introdução à Visão Computacional: Uma abordagem prática com Python e OpenCV**. São Paulo: Casa do Código, 2018.
- BASE64. Disponível em: <<https://pt.wikipedia.org/wiki/Base64>>. 14 mai. 2022. Acesso em: 09 dez 2023.
- BASSETT, Lindsay. **Introdução ao JSON: Um guia para JSON que vai direto ao ponto**. São Paulo: Novatec, 2019.
- DULDULAO, Devlin Basilan; VILLAFRANCA, Seiji Ralph. **Spring Boot and Angular**. Birmingham: Packt Publishing, 2022.
- FELTRIN, Fernando. **Visão Computacional em Python: Do básico às redes neurais artificiais: Versão Estendida**. [s.l.]: [s.n.], 2023.
- FREITAS, Felipe. **O que é e como funciona a resolução de imagem**. 14 jul. 2020. Disponível em: <<https://canaltech.com.br/software/resolucao-de-imagem-o-que-e/>>. Acesso em: 14 jan 2023.
- GONÇALVES, Marcelo. **Arquitetura de Software: Estilos e Padrões de Design**. 14 out. 2021. Disponível em: <<https://medium.com/@marcelomg21/arquitetura-de-software-estilos-e-padr%C3%B5es-de-design-50d62d684ef2>>. Acesso em: 24 nov 23.
- PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de Software: uma abordagem profissional**. 9. ed. Porto Alegre: AMGH, 2021.
- RUSSELL, Stuart J.; NORVIG, Peter. **Inteligência Artificial: uma abordagem moderna**. 4. ed. Rio de Janeiro: LTC, 2022.
- TESSERACT. Disponível em: <<https://tesseract-ocr.github.io/tessdoc/>>. Acesso em: 04 dez 2023.
- TESSERACT. Disponível em: <<https://github.com/tesseract-ocr/tesseract/tree/main>>. Acesso em: 04 dez 2023.
- TORRES, Fernando Esquirio. **Single Page Application**. São Paulo: Editora Senac, 2023.
- TORRES, Fernando Esquirio. **Desenvolvimento de API REST**. São Paulo: Editora Senac, 2021.
- WOHLIN, Claes *et al.* **Experimentation in Software Engineering**. Londres: Springer, 2012.

APÊNDICE A - FORMULÁRIO: Tecnologia & Cosméticos

Este formulário é parte integrante da dissertação de mestrado de Deise Dall’Agnol Severo, o qual é requisito parcial para formação no curso de Mestrado Profissional em Engenharia de Software da Universidade Federal do Pampa (UNIPAMPA).

O trabalho citado é referente ao desenvolvimento de uma aplicação web voltada para a identificação de produtos cosméticos.

As perguntas a seguir visam captar a opinião do público acerca da compra e utilização de produtos cosméticos.

TERMO DE CONFIDENCIALIDADE

Título do projeto: Utilização da tecnologia OCR para identificação de ingredientes cosméticos

Pesquisador responsável: Deise Dall’Agnol Severo

Campus/Curso: Alegrete/Mestrado Profissional em Engenharia de Software

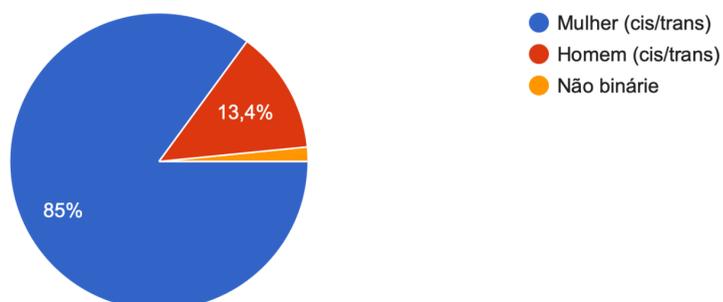
Local da coleta de dados: Google forms

Os pesquisadores do presente projeto se comprometem a preservar a privacidade e o anonimato dos sujeitos cujos dados serão coletados neste formulário do Google Forms. Concordam, igualmente, que estas informações serão utilizadas única e exclusivamente para execução do presente projeto. As informações somente poderão ser divulgadas preservando o anonimato dos sujeitos e serão mantidas em poder do responsável pela pesquisa, Prof. Pesquisador Cláudio Schepke por um período de 5 anos. Após este período, os dados serão destruídos.

Porto Alegre, 08 de junho de 2023.

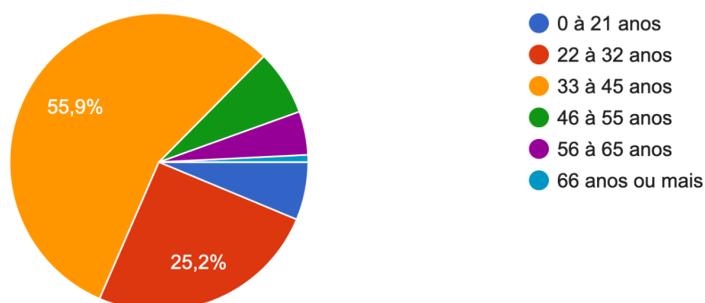
Você se considera:

254 respostas



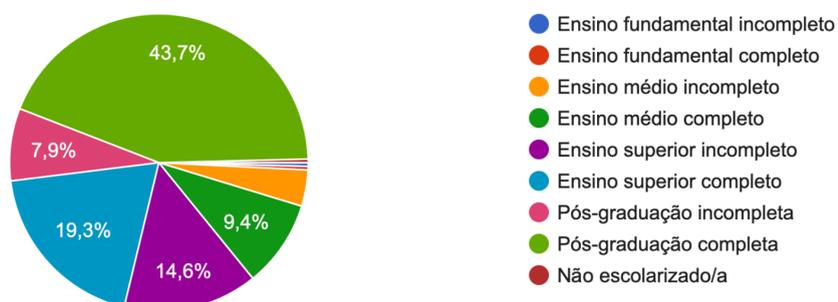
Qual a sua faixa etária?

254 respostas



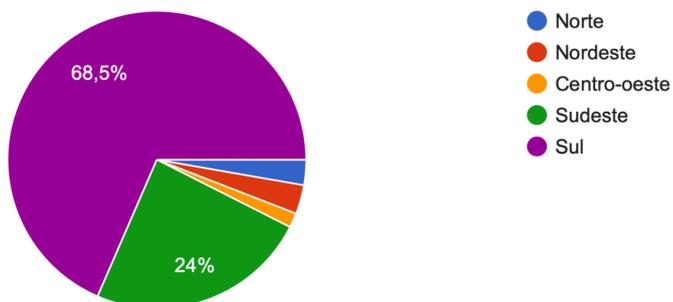
Qual a sua formação?

254 respostas



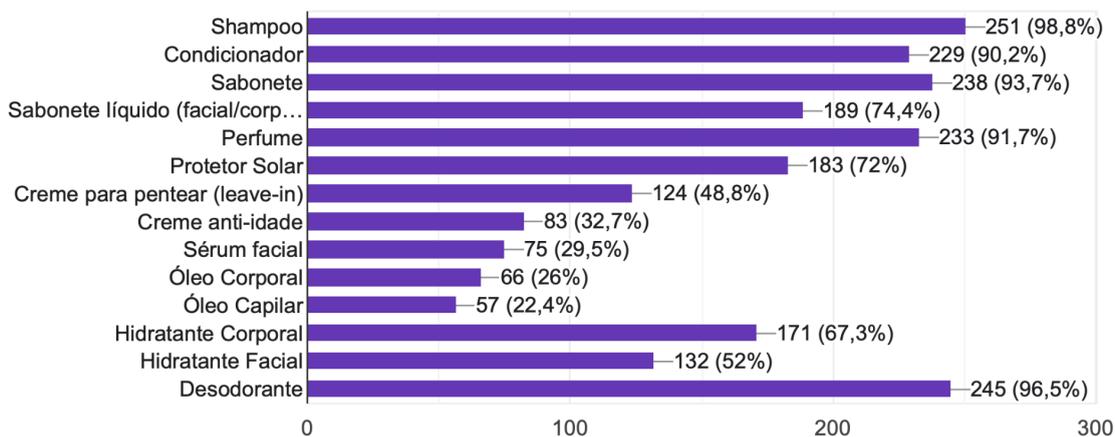
Em qual região você mora?

254 respostas



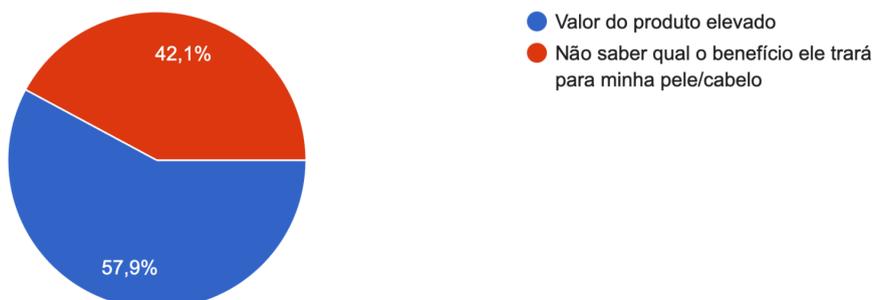
Qual cosmético você costuma usar?

254 respostas



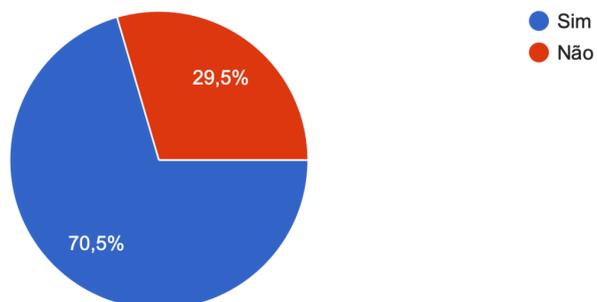
O que faz você deixar de comprar um produto cosmético?

254 respostas



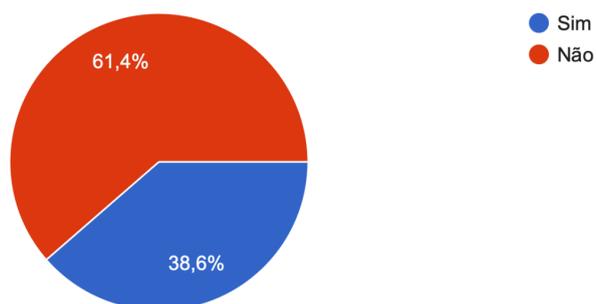
Saber a funcionalidade dos ingredientes da fórmula de um cosmético faria você comprá-lo mesmo que o valor do produto fosse caro?

254 respostas



Você costuma consultar com um profissional da beleza ou dermatologista para descobrir qual o melhor cosmético para sua pele/cabelo?

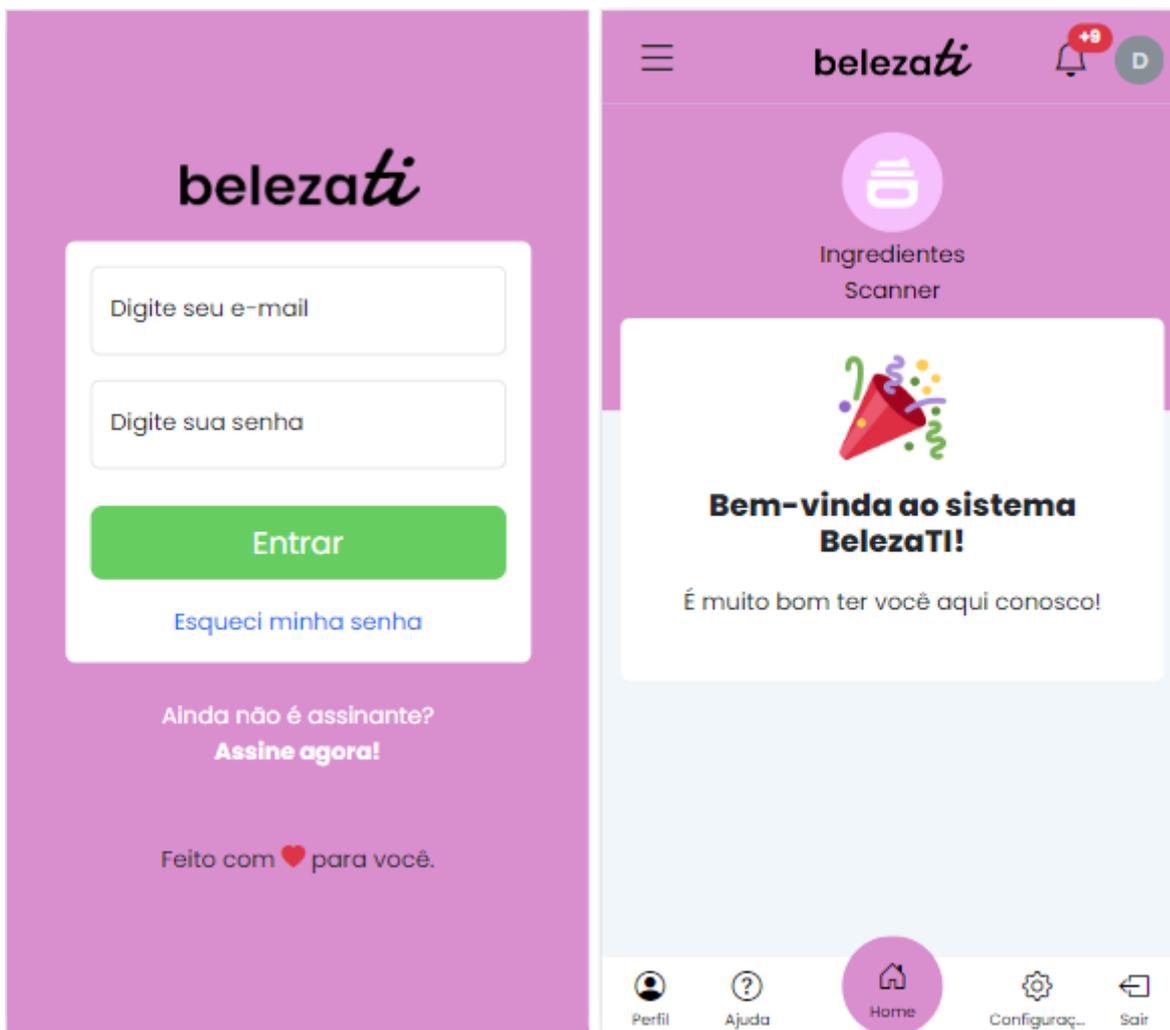
254 respostas



APÊNDICE B - DISPONIBILIDADE DO SISTEMA BELEZATI E TELAS

O sistema está disponível em: <https://app.belezati.com/>

Abaixo imagens das telas ilustrando a forma atual do módulo de escaneamento de ingredientes cosméticos do sistema Belezati:



← Ingredientes Scanner



← Ingredientes Scanner



Ingredientes:

Tire uma foto, envie uma imagem ou escreva a fórmula.

Analisar

Limpar

Ingredientes:

DIRECTIONS: Apply 3 to 4dregs of the product in the evening over the clean and dry face in upward movements for

Tire uma foto, envie uma imagem ou escreva a fórmula.

Analisar

Limpar

Ingredientes:

Tipo	Nome	Função
S	Água	Hidratante
I	Ácido Lático	Umectante
I	Propileno glicol	
I	Perfume	Perfumar

I Ingrediente S Substância

[Mais informações...](#)

ANEXO A - FUNÇÃO DOS INGREDIENTES COSMÉTICOS

FUNÇÃO (INGLÊS)	FUNÇÃO (PORTUGUÊS)	DEFINIÇÃO
ABRASIVE	ABRASIVOS	Substâncias adicionadas aos produtos cosméticos com o objetivo de remover materiais de diversas superfícies corporais, como auxiliares da limpeza dental mecânica ou com vista a aumentar o lustro.
ABSORBENT	ABSORVENTE	Substâncias adicionadas aos produtos cosméticos como meio de dispersão de substâncias dissolvidas em água ou em gorduras ou finamente divididas.
ANTICAKING	ANTIFERMENTATIVA	Permite o fluxo livre de partículas sólidas e, assim, evita a aglomeração de produtos cosméticos em pó em nódulos ou massas duras
ANTICORROSIVE	ANTICORROSIVO	Impede a corrosão da embalagem
ANTI DANDRUFF	ANTICASPA	Ajuda a controlar a caspa
ANTIFOAMING	ANTIESPUMANTE	Suprime espuma durante a fabricação ou reduz a tendência de produtos acabados para gerar espuma
ANTIMICROBIAL	ANTIMICROBIANO	Ajuda a controlar o crescimento de microrganismos sobre a pele
ANTIOXIDANT	ANTIOXIDANTE	Inibe reações promovidas pelo oxigênio, evitando a oxidação e rancificação
ANTIPERSPIRANT	ANTIPERSPIRANTE	Reduz a transpiração
ANTIPLAQUE	ANTIPLACA	Ajuda a proteger contra a placa
ANTI SEBORRHOEIC	ANTI SEBORRÉICO	Ajuda a controlar a produção de sebo
ANTISTATIC	ANTIESTÁTICO	Reduz a eletricidade estática por carga elétrica de neutralização sobre uma superfície
ASTRINGENT	ADSTRINGENTE	Contraí a pele
BINDING	LIGANTE	Fornece coesão em cosméticos
BLEACHING	BRANQUEADOR	Clareia a sombra de cabelo ou da pele
BUFFERING	AGENTE TAMPÃO	Estabiliza o pH dos produtos cosméticos
BULKING	AGENTE DE VOLUME	Reduz a densidade a granel de produtos cosméticos
CHELATING	QUELANTE	Substâncias adicionadas aos produtos cosméticos com o objetivo de formar complexos com íons metálicos susceptíveis de afetar a estabilidade e/ou o aspecto dos cosméticos.

CLEANSING	LIMPADORES	Ajuda a manter a superfície do corpo limpa
COSMETIC COLORANT	CORANTES COSMÉTICOS	Substâncias adicionadas aos produtos cosméticos com o objetivo de lhes conferir uma coloração e/ou induzir a coloração da pele e/ou das respectivas excrescências. Todos os corantes enumerados são substâncias da lista positiva de corantes (anexo IV da Diretiva dos produtos cosméticos)
DENATURANT	DESNATURANTES	Substâncias adicionadas, de um modo geral, a produtos cosméticos que contenham álcool etílico, de modo a torná-los impróprios para beber.
DEODORANT	DESODORIZANTE	Substâncias adicionadas aos produtos cosméticos com o objetivo de reduzir ou ocultar odores corporais desagradáveis
DEPILATORY	DEPILATÓRIO	Remove pêlos indesejáveis
DETANGLING	DESEMBARAÇANTE	Reduz ou elimina entrelaçamento cabelo devido a alteração dos cabelos ou danos de superfície e, assim, ajuda a pentear
EMOLLIENT	EMOLIENTE	Amacia e suaviza a pele
EMULSIFYING	EMULSIONANTE	Promove a formação de misturas íntimas de líquidos não miscíveis, alterando a tensão interfacial
EMULSION STABILISING	ESTABILIZADOR DE EMULSÃO	Ajuda o processo de emulsificação e melhora a estabilidade da emulsão e prazo de validade
FILM FORMING	FORMADOR DE FILME	Produz, mediante requerimento, um filme contínuo sobre a pele, cabelo e unhas
FLAVOURING	AROMATIZANTES	Dá sabor ao produto cosmético
FOAM BOOSTING	IMPULSIONADOR DE ESPUMA	Melhora a qualidade da espuma produzida por um sistema através do aumento de um ou mais das propriedades seguintes: Textura volume, e / ou estabilidade
FOAMING	ESPUMANTE	Pequenas bolhas de ar ou outro gás no interior de um pequeno volume de líquido, modificando a tensão superficial do líquido
GEL FORMING	GELIFICANTES	Dá a consistência de um gel (uma preparação semi-sólido com um pouco de elasticidade) para uma preparação líquida
HAIR CONDITIONING	CONDICIONANTE DE CABELO	Deixa o cabelo mais fácil de pentear, flexível, macia e brilhante e / ou confere volume, leveza, brilho, etc
HAIR DYEING	COLORANTE DE CABELOS	Substâncias adicionadas aos produtos cosméticos com vista à coloração do cabelo
HAIR FIXING	FIXADOR DE CABELO	Permite o controle físico do estilo de cabelo
HAIR WAVING OR STRAIGHTENING	ONDULADOR OU ALISADOR DE CABELO	Modifica a estrutura química do cabelo, permitindo-lhe ser ajustado no estilo requerido
HUMECTANT	UMECTANTE	Mantém e retém a umidade
HYDROTROPE	HIDROTRÓPICO	Aumenta a solubilidade da substância que é apenas ligeiramente solúvel em água

KERATOLYTIC	QUERATOLÍTICO	Ajuda a eliminar as células mortas do estrato córneo
MASKING	MASCARANTE	Reduz ou inibe o odor ou o sabor de base do produto
MOISTURISING	HIDRATANTE	Aumenta o teor de água da pele e ajuda a mantê-la macia e suave
NAIL CONDITIONING	CONDICIONADOR DE UNHAS	Melhora as características estéticas da unha
OPACIFYING	OPACIFICANTES	Reduz a transparência ou translucidez dos cosméticos
ORAL CARE	CUIDADO ORAL	Fornece efeitos cosméticos para a cavidade oral, por exemplo, limpeza, desodorização, protegendo
OXIDISING	OXIDANTES	Altera a natureza química da substância pela adição de um outro átomo de oxigénio ou a remoção de hidrogénio
PEARLESCENT	PEARLESCENT	Dá uma aparência mascarada a cosméticos
PERFUMING	PERFUMES	Usada para perfumes e matérias-primas aromáticas (secção II)
PLASTICISER	PLASTIFICANTE	Suaviza e faz outra substância flexível que de outra forma não poderia ser facilmente deformada, espalhar ou trabalhados
PRESERVATIVE	CONSERVANTE	Inibe primariamente o desenvolvimento de microrganismos em cosméticos. Todos os conservantes listados são substâncias na lista positiva de conservantes (anexo VI da Directiva Cosméticos)
PROPELLANT	PROPULSOR/PROPELENTE	Gera pressão numa embalagem de aerossol, expelindo o conteúdo quando a válvula é aberta. Alguns gases propulsores liquefeitos podem actuar como solventes
REDUCING	REDUTOR	Altera a natureza química da substância pela adição de um outro átomo de hidrogénio ou remoção de oxigénio
REFATTING	SOBREENGORDURANTE	Reabastece os lípidios do cabelo ou das camadas superiores da pele
REFRESHING	REFRESCANTE	Confere uma frescura agradável para a pele
SKIN CONDITIONING	CONDICIONADOR DA PELE	Mantém a pele em boas condições
SKIN PROTECTION	PROTETOR DA PELE	Ajuda a evitar efeitos prejudiciais para a pele de fatores externos
SMOOTHING	SUAVIZANTE	Procura alcançar uma superfície uniforme da pele, diminuindo a rugosidade ou irregularidades
SOLVENT	SOLVENTE	Dissolve outras substâncias
SOOTHING	CALMANTE	Ajuda a aliviar o desconforto da pele ou do couro cabeludo
STABILISING	ESTABILIZANTE	Melhora ingredientes ou a estabilidade da formulação e prazo de validade

SURFACTANT	SURFACTANTE/TENSOATIVO	Diminui a tensão superficial da cosmética, bem como auxiliares de distribuição uniforme do produto, quando usado
TANNING	BRONZEADOR	Escurece a pele, com ou sem exposição aos UV
TONIC	TÔNICO	Produz uma sensação de bem-estar na pele e cabelo
UV ABSORBER	ABSORVEDOR DE RAIOS UV	Protege o produto cosmético dos efeitos da luz UV
UV FILTER	FILTRO UV	Filtrar certos raios UV, a fim de proteger a pele ou os cabelos contra os efeitos nocivos destas radiações. Todos os filtros UV listados são substâncias na lista positiva de filtros de UV (anexo VII da Directiva Cosméticos)
VISCOSITY CONTROLLING	REGULADOR DE VISCOSIDADE	Aumenta ou diminui a viscosidade dos cosméticos