

# Uso de aplicação web no auxílio da escolha de lentes no balcão de uma óptica

Rodrigo Maciel Moreira<sup>1</sup>, Amir Tauille<sup>2</sup>

<sup>1</sup>Discente de Tecnologia em Análise e Desenvolvimento de Sistemas – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul - Campus Avançado Veranópolis - RS - Brasil

<sup>2</sup>Docente de Tecnologia em Análise e Desenvolvimento de Sistemas – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul - Campus Avançado Veranópolis - RS - Brasil

rodrigo.mm82@hotmail.com, amir.tauille@veranopolis.ifrs.edu.br

**Resumo.** Este artigo demonstrou as etapas de estudo e desenvolvimento de uma aplicação web, que teve por objetivo ser um facilitador do profissional do ramo óptico. O estudo abrangeu os cálculos necessários para obter o diâmetro mínimo de uma lente, transposição de dioptrias, além de cálculos de dioptria de perto e adição para lentes multifocais. Através de consulta com público-alvo, foi possível perceber que a aplicação agiliza consideravelmente a obtenção destas informações. Estes dados são disponibilizados pelos laboratórios fabricantes de lentes, mas não com a rapidez necessária. Para este desenvolvimento, foi utilizada a linguagem de programação JavaScript, com o auxílio de framework Vue.JS e da biblioteca Quasar.

**Abstract.** This article demonstrated the stages of study and development of a web application, which aims to be a facilitator of the optical professional. The study covers the calculations needed to obtain the minimum diameter of a lens, diopter transposition, as well as diopter calculations closely and addition to multifocal lenses. Through consultation with a target audience, it was possible to realize that the application considerably speeds up the obtaining of this information. This data is made available by the lens manufacturers laboratories, but not as quickly as necessary. For this development, the JavaScript programming language will be used, with the help of Vue.JS framework and Quasar library.

## 1.Introdução

Com o passar dos anos, a indústria de lentes oftálmicas cresceu muito. Tanto na infinidade de opções de materiais e design de lentes, quanto na tecnologia empregada na produção e tomadas de medidas. Entretanto, nem todos os profissionais do ramo óptico tem acesso a toda esta modernidade.

Sendo assim, este artigo forneceu elementos que justificaram o desenvolvimento de uma aplicação web para facilitar o trabalho do atendente no dia a dia com ferramentas que auxiliem em diversos cálculos que até então só eram acessíveis em laboratórios. Os mesmos fornecem o resultado destes cálculos, contudo não com a mesma agilidade, necessária para quem está na frente do cliente no balcão da óptica.

Estes cálculos são necessários para definir quais os tipos de lentes que podem ser selecionadas ao cliente final. Com esses dados em mãos será possível elencar, dentre as opções disponíveis, as que podem dar um melhor resultado ao cliente, tanto em qualidade visual quanto estético, mas sem esquecer que estas informações também são necessárias para, tanto diminuir o erro na escolha das lentes, quanto optar por uma que não possua diâmetro suficiente.

Para o desenvolvimento do sistema, foram utilizadas como ferramentas, a linguagem *JavaScript* e o *framework* (estruturas compostas por um conjunto de códigos genéricos que permite o desenvolvimento de sistemas e aplicações) *Vue.JS* além da biblioteca *Quasar*. Esta escolha se deve pelo conhecimento prévio do desenvolvedor, tanto da linguagem de programação, quanto do *framework* e a biblioteca.

O objetivo é facilitar a tomada de decisão sobre o material das lentes, a correta transposição de lente com dioptria cilíndrica positivo para negativo. Além de possuir cálculo de diâmetro da lente, cálculo de adição e dioptria de perto.

Segundo o Dicionário Michaelis (2015), dioptria é “unidade de medida de convergência de uma lente, que equivale ao inverso da distância focal expressa em metros”, comumente, mas erroneamente chamado de grau. Grau, por sinal, é a medida do eixo de inclinação da dioptria cilíndrica (astigmatismo).

## **2.Fundamentação Teórica**

### **2.1. Tecnologias**

*JavaScript* (JS) é uma linguagem de programação para ambiente web. Ao passo que este ambiente evoluiu, o JS se tornou uma linguagem otimizada para criar vários conteúdos em HTML, CSS e várias páginas e aplicativos web [Crockford 2008, apud Kim e Lee 2018] e [Thompsons 2017, apud Kim e Lee 2018]. A partir disto permitiu-se ter em um único código-fonte, gerenciar tanto o conteúdo exibido, quanto a estilização e o layout.

JS é uma linguagem de programação orientada a objetos, mas diferentemente de Java e C++, ela é fracamente tipada [Wagner e Diaconescu 2017]. Isso facilita muito no reuso de componentes, pois ao cria-lo não é necessário definir o tipo, apenas designar um valor. O *framework* *Vue.JS*, possui uma grande quantidade de métodos. Atualmente os *browsers* (navegadores) já possuem *engine* (são softwares que contém diversas ferramentas/bibliotecas) para suporte do JS, não sendo necessários *plugins* (programa, extensão ou ferramenta que pode ser adicionada no programa principal) adicionais [Chatzimpampas, et al. 2019].

Segundo Rauschmayer (2015) JS além de gratuito, é útil, possui boas ferramentas. Rauschmayer (2015) também disse que, JavaScript pode não ser o ápice da elegância. “No entanto, é uma linguagem muito flexível, possui um núcleo razoavelmente elegante e permite usar uma mistura de programação orientada a objetos e programação funcional.”

### **2.2. Óptica Oftálmica**

De acordo com Dias (2011), um vendedor óptico deve possuir, não só uma capacidade de diálogo fácil, mas também conhecimento técnico da ciência óptica. Nisto, inclui-se o conhecimento das diversas opções de materiais de lentes, como resinas, policarbonato e

minerais, como também das corretas tomadas de medidas. Também é necessária a escolha apropriada da armação, conforme formato de rosto e tipo de dioptria prescrita pelo oftalmologista.

Segundo Sonoda (2020) transposição “é mudar a designações das dioptrias, mantendo seu poder” (Tabela 1), ou seja, alterar a dioptria prescrito pelo oftalmologista, mas sem alterar o poder da lente. Ele também explica que, basta somar o valor do cilíndrico ao esférico, depois inverter o sinal do cilíndrico. Por último somar 90° ao eixo menor que 90°, ou diminuir se for maior ou igual a 90°, sendo que seu valor deve ficar entre 0° e 180°. Cálculo geralmente utilizado quando o cilíndrico for positivo, de modo que:

**Tabela 1 Transposição**

Dioptria da receita	Esf. -2,00 Cil. 1,00 Eixo 110°
Dioptria transposta	Esf. 1,00 Cil. -1,00 Eixo 20°

Sonoda (2020) também afirma que, há casos em que o receituário possui distintas dioptrias para longe e para perto. Esta diferença, é denominada adição (ad). Para calcular, basta subtrair do valor de perto o que foi informado para longe (Tabela 2). Também é possível o inverso, quando temos somente o esférico de longe e a adição, para encontrar a dioptria de perto. Este resultado é obtido através da soma da dioptria de longe com a adição (Tabela 3).

**Tabela 2 Cálculo de adição**

Longe	Esf. -2,00 Cil. -1,00 Eixo 110°	Ad ?
Perto	Esf. +1,00 Cil. -1,00 Eixo 110°	

Ex.: Adição = perto - longe = 1,00 - (-2,00) = 1,00 + 2,00 = +3,00

**Tabela 3 Cálculo para dioptria de perto**

Longe	Esf. -2,00 Cil. -1,00 Eixo 110°	Ad +3,00
Perto	?	

Ex.: Perto = longe + adição = -2,00 + 3,00 = +1,00

Atualmente, há uma crescente busca por armações grandes, e há casos em que a loja não possui técnico qualificado. Isto pode acarretar uma escolha indevida, pois lente e armação, podem possuir uma inviabilidade técnica na montagem [Sonoda, 2020].

Para garantir uma escolha adequada, Sonoda (2020), recomenda o cálculo:

$$DMM = (P + A) - (\text{Menor DNP} \times 2) + DM + MS$$

Onde:

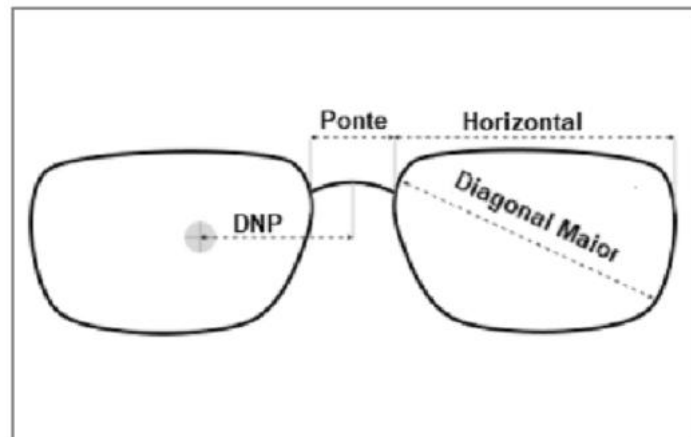
P = Ponte (mm)

A = Aro ou Horizontal (mm)

DNP = Distância Naso-Pupilar (mm)

DM = Diagonal Maior (mm)

MS = Margem de Segurança (normalmente 2mm)

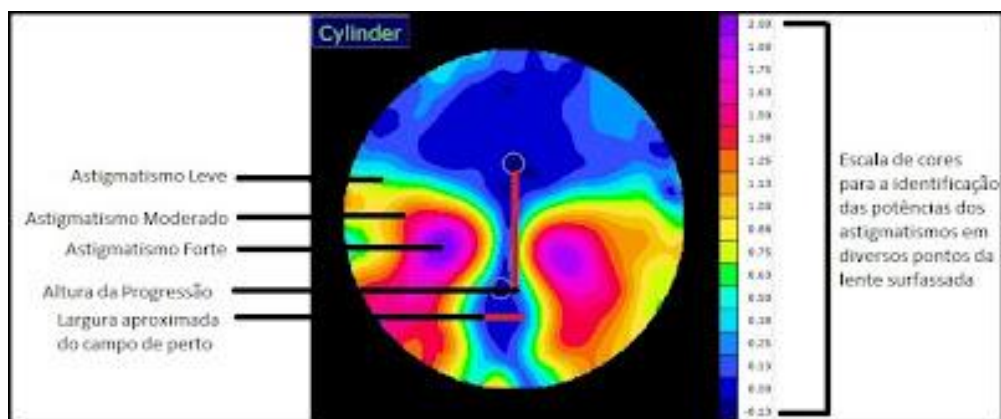


**Figura 1 Modelo de medidas**

No momento que a lente foi vendida, um passo muito importante é a tomada de medidas. Uma lente de qualidade reconhecida e uma armação bem escolhida são importantíssimas para obter um bom resultado. Entretanto, se as medidas tomadas junto ao cliente, forem feitas de maneira descuidada, podemos perder o trabalho feito até então.

“Nos pedidos de lentes surfassadas é necessário anotar na ordem de serviço o valor em milímetros da ponte, aro, diagonal maior e vertical da armação” [Maciel 2013]. Além destas medidas para a fabricação das lentes, são necessárias também a DNP (distância naso-pupilar) e CO ou ACO (altura de centro óptico). A DNP é a tomada de medidas entre o centro do nariz até o centro de cada pupila, resultando em dois valores; uma para o olho esquerdo e outro para o olho direito. Já a ACO também é medida a partir do centro da pupila até a base inferior da armação escolhida. Estas medidas são necessárias tanto para as multifocais, bifocais e monofocais.

Com essas medidas em mãos, é possível produzir uma lente personalizada para o cliente. Também neste momento é necessário ter muita atenção, já que a tomada de medidas incorretas ou uma transcrição destas medidas, ao laboratório de produção de lentes, com erros, podem prejudicar a adaptação do usuário das mesmas. Caso um erro como esse ocorra, poderemos alterar o ponto focal correto deste usuário, induzindo um astigmatismo indevido a ele (Figura 2).



**Figura 2 Escala de cores, onde cada uma revela a indução de astigmatismo em uma lente multifocal - Fonte: [blogarama.com](http://blogarama.com)**

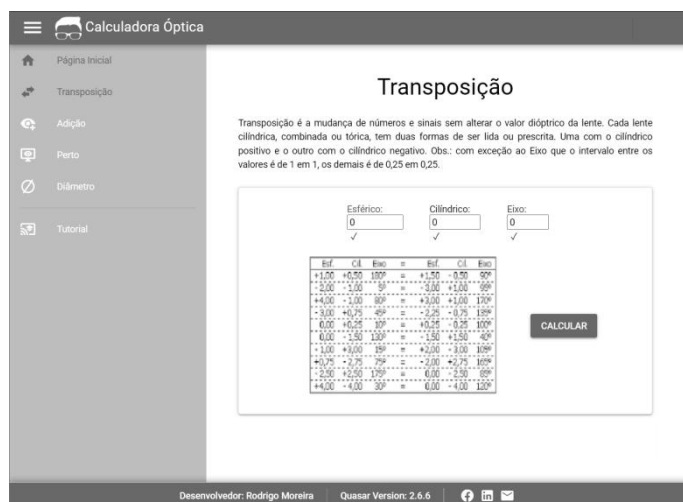
Nos casos das lentes monofocais ou visão simples também são importantes, mas por outras razões. Este tipo de lente possui duas apresentações: surfacada e pronta ou semiacabada. Como dito anteriormente, nas lentes surfacadas são necessárias todas as medidas utilizadas para a fabricação. Já nas prontas, estas medidas são importantíssimas para a verificação do diâmetro do bloco necessário. Em alguns casos, como nas lentes onde a dioptria esférica é positiva, este diâmetro é de até 65mm. Isto impossibilita o encaixe em armações um pouco maiores e até o tipo de acabamento.

### 3.Métodos

Com a utilização da linguagem de programação JS, foi criada uma aplicação web no formato *single-page application* (aplicação em tela única) (Imagem 1), nela constam abas voltadas para cada cálculo, com imagens indicativas de cada medida a ser inserida (Imagem 2).



**Imagem 1 Tela inicial da web application**



**Imagem 2 Modelo de exibição das telas de medidas e menu lateral.**

Com o objetivo de minimizar ao máximo os erros de inserção dados, será utilizado um campo onde será possível selecionar o valor da medida, mas também será possível inserir manualmente. A partir destas informações, serão realizados os cálculos

necessários, sendo eles extraídos de sites de alguns ópticos brasileiros, como os professores Paulus Maciel e Ney Dias e do livro do professor Sonoda (2020).

Além disso, os campos possuem validação de informação inserida, com a exibição de um  para inserções inválidas e um  nas válidas. Caso pelo menos um dos campos for inválido, o botão que dispara o método de cálculo, fica desabilitado. Cada campo possui um *tooltip* (dica), para facilitar seu preenchimento.

Após o desenvolvimento da aplicação, ela foi disponibilizada em duas ópticas da região, com intuito de testar as funcionalidades e descobrir possíveis *bugs*. Com a primeira etapa de testes finalizada, foram corrigidas as falhas e/ou melhorias de layout. Em seguida, realizou-se uma nova etapa de testes, para validação das melhorias.

### 3.1. Diagrama de Classes

Diagrama retratando as classes, atributos e métodos utilizados no projeto.

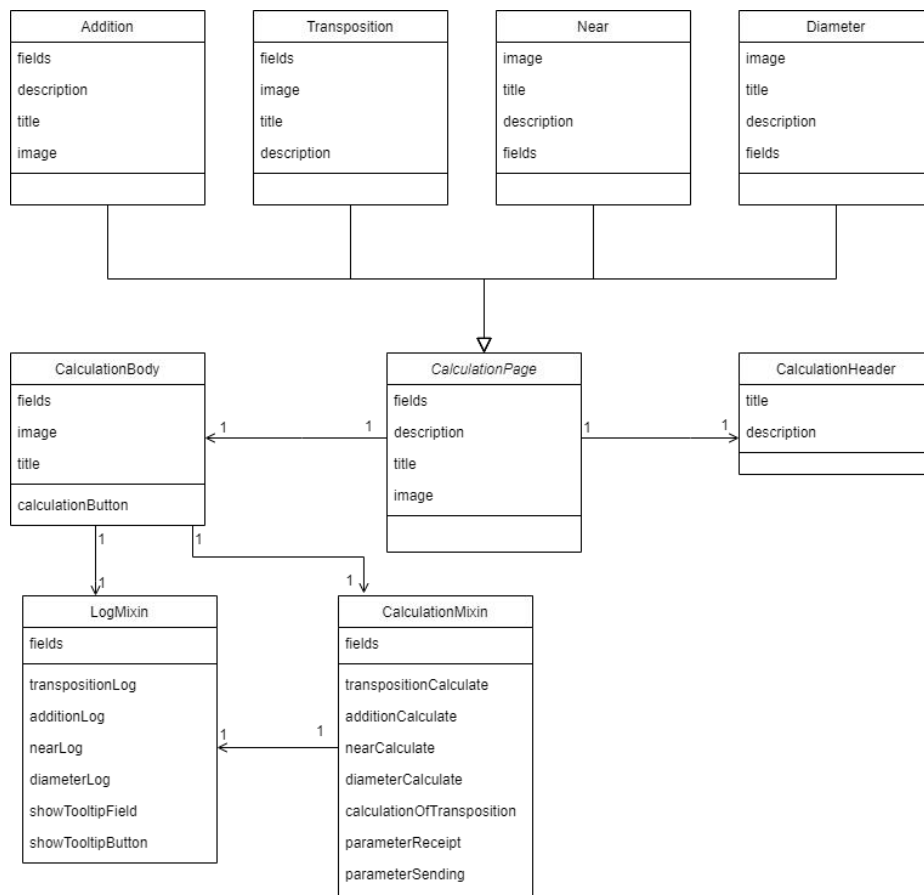
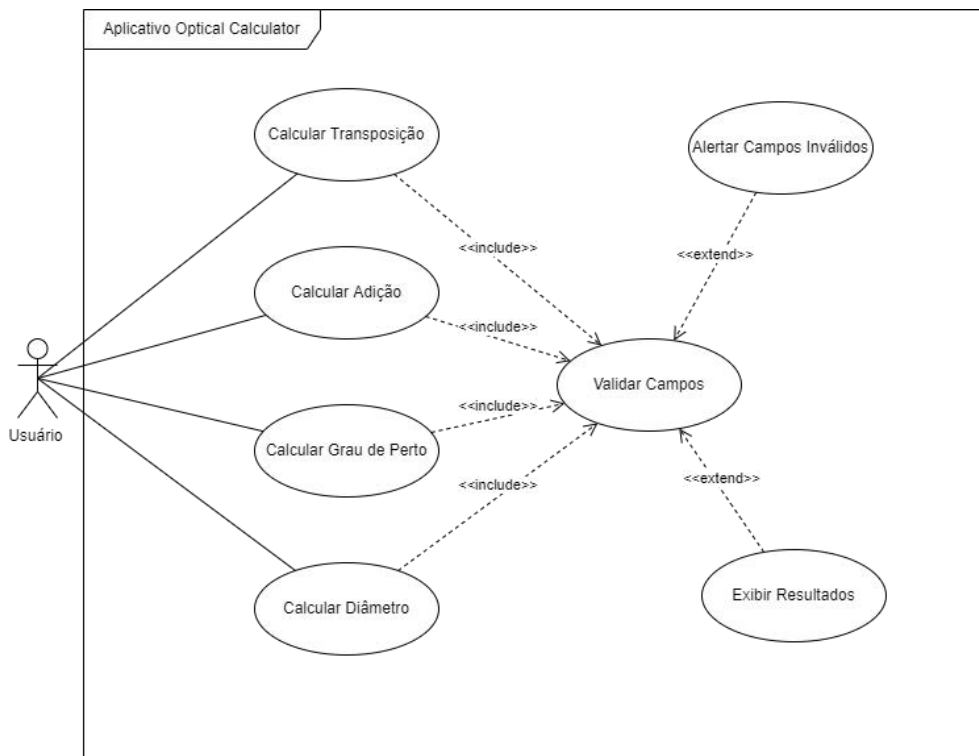


Figura 3 Diagrama de Classes, Modelagem Conceitual

### 3.2. Diagrama de Caso de Uso

Diagrama retratando os casos de uso disponíveis e seus possíveis retornos.



**Figura 4 Diagrama de caso de uso da aplicação**

**Tabela 4 Documentação do diagrama de casos de uso “Calcular Transposição”**

Nome do Caso de Uso	Calcular Transposição
Ator	Usuário
Pré-condição	Preencher os campos com dados válidos
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Acessar a tela do cálculo	
2. Inserir valores nos campos	
	3. Validar os valores inseridos
	4. Exibir resultado
Fluxo Alternativo	
Ações do Ator	Ações do Sistema
	1. Alertar campo inválido
	2. Desabilita botão de calcular

**Tabela 5 Documentação do diagrama de casos de uso “Calcular Adição”**

Nome do Caso de Uso	Calcular Adição
Ator	Usuário
Pré-condição	Preencher os campos com dados válidos
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Acessar a tela do cálculo	
2. Inserir valores nos campos	
	3. Validar os valores inseridos
	4. Exibir resultado
Fluxo Alternativo	
Ações do Ator	Ações do Sistema
	1. Alertar campo inválido
	2. Desabilita botão de calcular

**Tabela 6 Documentação do diagrama de casos de uso “Calcular Grau de Perto”**

Nome do Caso de Uso	Calcular Grau de Perto
Ator	Usuário
Pré-condição	Preencher os campos com dados válidos
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Acessar a tela do cálculo	
2. Inserir valores nos campos	
	3. Validar os valores inseridos
	4. Exibir resultado
Fluxo Alternativo	
Ações do Ator	Ações do Sistema
	1. Alertar campo inválido
	2. Desabilita botão de calcular



**Tabela 7 Documentação do diagrama de casos de uso “Calcular Diâmetro”**

Nome do Caso de Uso	Calcular Diâmetro
Ator	Usuário
Pré-condição	Preencher os campos com dados válidos
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Acessar a tela do cálculo	
2. Inserir valores nos campos	
	3. Validar os valores inseridos
	4. Exibir resultado
Fluxo Alternativo	
Ações do Ator	Ações do Sistema
	1. Alertar campo inválido
	2. Desabilita botão de calcular

### 3.3. Criação de classes

Foi desenvolvida a classe *MainLayout* (layout principal), com a chamada dos componentes utilizados em todas as telas, sendo eles: *LeftDrawer* (menu lateral), *CalculationRouterView* (redirecionamento para as classes de objetos) e *CalculationFooter* (rodapé). Além disso, é utilizada a tag nativa do *Quasar* para a criação da barra de ferramentas.

Na barra de ferramentas, foi inserido um botão para ocultar/exibir o menu lateral, além da logo e nome da aplicação. Também possui, no lado superior direito, um campo onde é exibido data e hora (não disponível na exibição em mobile).

Com a utilização de reuso de código, foram criadas as classes responsáveis somente pelos objetos e seus atributos, sendo elas *Addition* (cálculo de adição), *Diameter* (cálculo de diâmetro), *Near* (cálculo de dioptria de perto), *Transposition* (cálculo de transposição de dioptria). Dentro de cada uma delas há a chamada da classe de exibição em tela, *CalculationPage*.

Na classe *LeftDrawer*, exibimos os itens responsáveis pelo redirecionamento às páginas. Cada linha possui um ícone, um título, além do *path* (caminho). A partir deste é chamada a classe *CalculationLinks*, que por sua vez é responsável pela organização e estilização dos links.

A *CalculationFooter* é responsável por popular e alinhar as informações do rodapé. Nela encontram-se as informações sobre que o nome do desenvolvedor, suas redes sociais e email de contato.

Já na classe *CalculationPage* é onde são recebidas as *props* (atributos) vindos do objeto. Nela há a chamada de outras classes, que são responsáveis pela exibição de cada

componente: *CalculationHeader* (cabeçalho), *CalculationBody* (corpo da página), sendo que cada uma recebe suas próprias *props*.

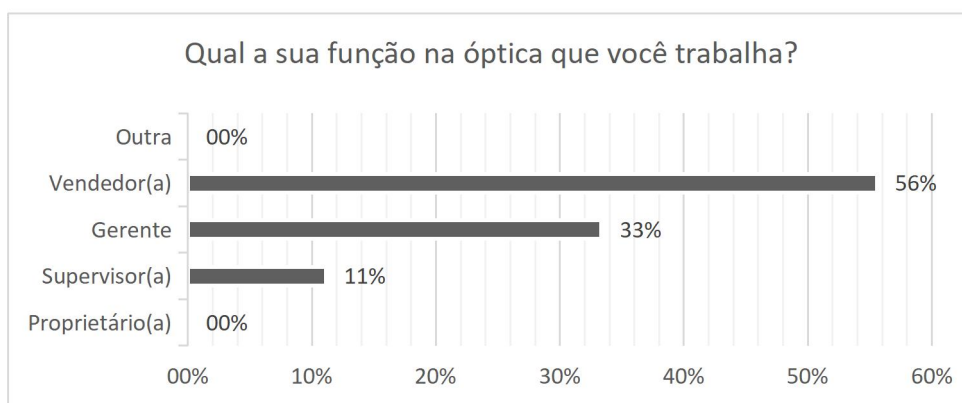
A classe *CalculationBody* exibe o objeto de cada página, sendo cada atributo, um campo para inserção ou seleção de valores, além de seu nome e um validador de valores. Também mostra uma imagem ilustrativa do tipo de cálculo a ser feito, um botão que é ativado, caso os valores sejam válidos e um campo de resultado. No caso do campo que exibe o resultado, há a exceção da página de cálculo de transposição, pois nela, os valores são alterados diretamente em seus campos de origem.

#### 4. Resultados

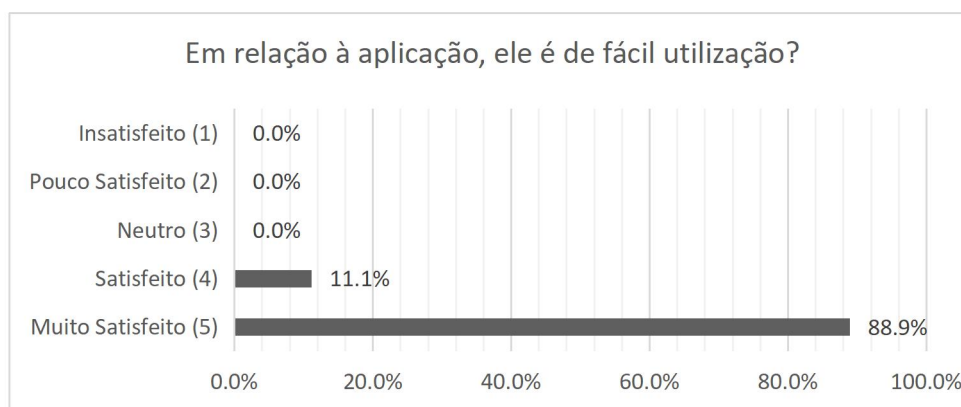
For realizada uma pesquisa online, onde nela constam algumas perguntas que teve como objetivo avaliar a aceitação e a usabilidade do público-alvo, onde obtivemos um total de nove pessoas respondendo esta pesquisa.

No total foram realizadas oito perguntas, sendo sete delas obrigatórias e de seleção de resposta. Uma última pergunta foi realizada, sendo essa descritiva e não obrigatória: “Descreva abaixo, dicas e sugestões para a melhoria de interface e usabilidade da aplicação”. Esta por sua vez, não foram adicionadas respostas.

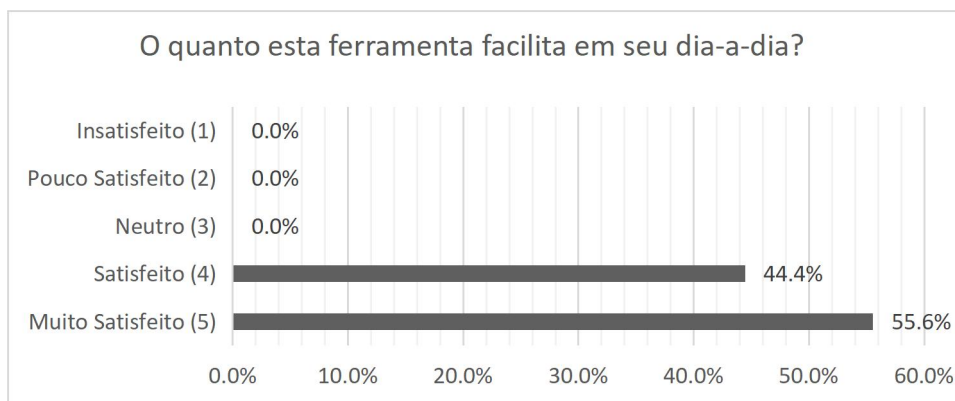
**Gráfico 1 Profissão dos pesquisados**



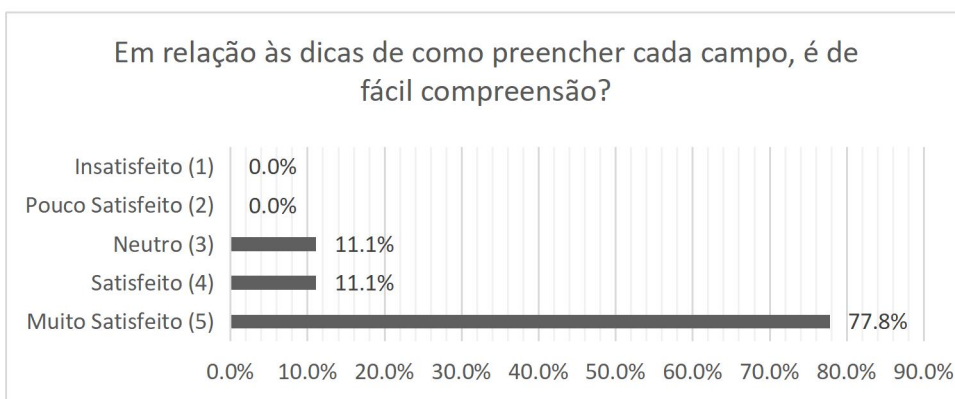
**Gráfico 2 Satisfação dos pesquisados**



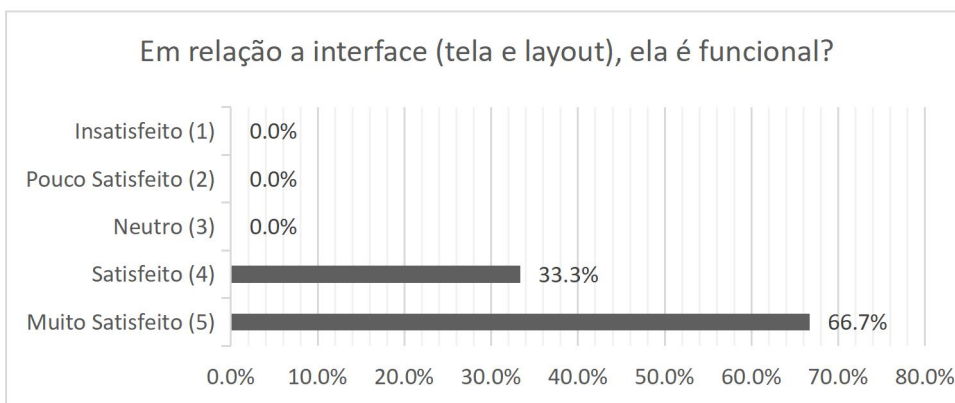
**Gráfico 3 Nível de facilitação aos pesquisados**



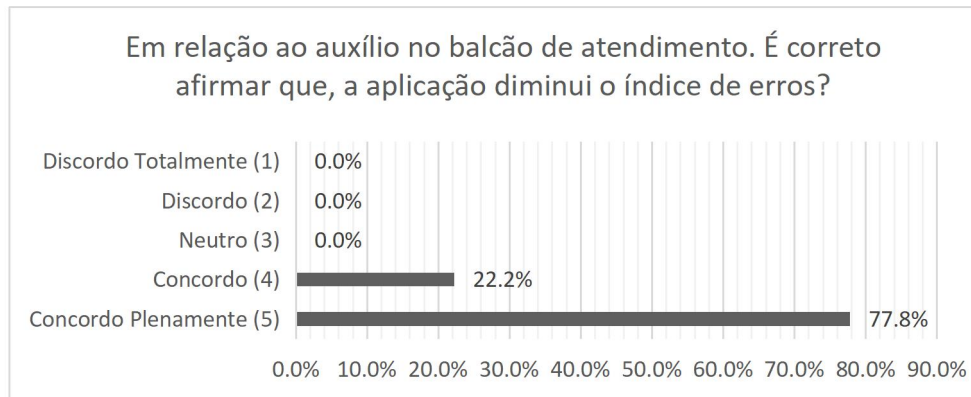
**Gráfico 4 Nível de compreensão das dicas, pelos pesquisados**



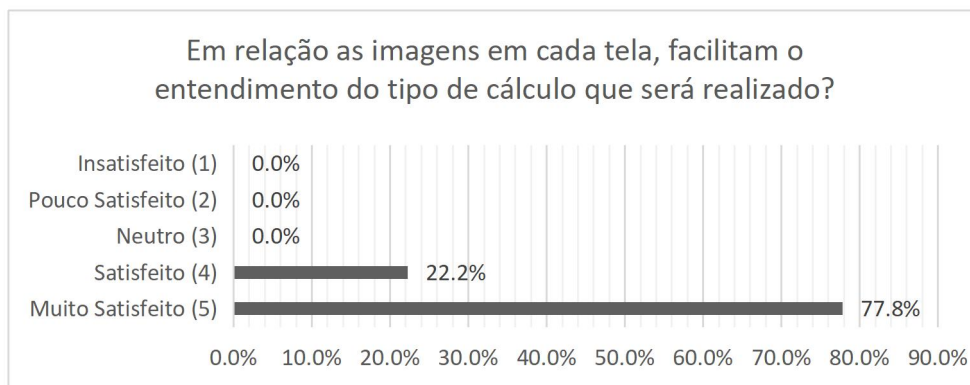
**Gráfico 5 Satisfação dos pesquisados em relação à interface**



**Gráfico 6 Diminuição de erros com o uso da aplicação**



**Gráfico 7 O quanto as ilustrações auxiliam na aplicação**



## 5. Considerações Finais

Este artigo apresentou uma pesquisa e o desenvolvimento de um sistema para cálculos destinados ao ramo óptico varejista e as tecnologias que foram empregadas nesta solução computacional. Nela, foram abordados os cálculos, como utilizá-los na solução e o que pode acarretar um cálculo feito de modo equivocado.

Através desta pesquisa foi possível desenvolver uma aplicação web no formato *single-page application*. Esta aplicação possui uma gama de cálculos necessárias para uma melhor escolha de lentes ao cliente. Cálculos estes que antes eram solicitados ao laboratório óptico, podendo levar até um dia para serem enviados os resultados, agora podem ser realizados em minutos.

Dentre estes cálculos, podemos exemplificar aquele utilizado no diâmetro da lente. Lentes conhecidas como semiacabadas possuem um diâmetro específico, de acordo com a indicação. Caso seja para compensação de miopia, possuem 70mm, caso seja para hipermetropia, 65mm.

Levando em consideração estes diâmetros pré definidos, um usuário que pretende adquirir seus óculos e esta armação possuir aro (A) de 55mm, diagonal maior (DM) de 57mm, ponte (P) de 17mm e este usuário possuir DNP de 30 e 31,5:  $(P + A) - (\text{Menor DNP} \times 2) + DM + MS = (17 + 55) - (30 \times 2) + 57 + 2 = 71\text{mm}$ . Neste caso, independentemente do tipo de ametropia (distúrbio visual decorrente da focalização inadequada da luz que chega à retina) que este usuário sofre, as lentes semiacabadas serão inadequadas. Com isso, o indicado é fabricar uma lente de forma personalizada, levando em consideração a ametropia e o diâmetro e as características da armação

escolhida. Esta opção resulta em maior investimento, entretanto permite que a lente correta seja escolhida evitando que o cliente pague por algo que não foi observado na compra e que a loja não precise arcar com a diferença de custo da lente.

Está sendo estudada uma série de melhorias futuras, como histórico de clientes e suas respectivas dioptrias e medidas utilizadas, além de cálculo de espessura de borda. Estas melhorias além de garantir uma posterior consulta a informações anteriores dos clientes, também auxiliaram numa melhor escolha de lente, como um material mais e com maior qualidade visual.

## 6. Referências

- Ali Gunawan, "Selection of Open Source Database Management for System Development Using Analytic Hierarchy Process Method in PT. XYZ". 2020 International Conference on Information Management and Technology (ICIMTech), 2020.
- Chatzimparmpas, Angelos; Bibi, Stamatia; Zozas, Ioannis; Kerren, Andreas. (2019). "Analyzing the Evolution of Javascript Applications". In Proceedings of the 14th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE.
- Dias, Ney. (2011) "126 Dicas do Ney Dias", <https://sites.google.com/site/neydiasopticaoftalmica/126-dicas-do-ney-dias>, Junho.
- Kim, Jaehyun. Lee, Yangsun. (2018). "A Study on Abstract Syntax Treefor Development of a JavaScript Compilers". In International Journal of Grid and Distributed Computing, Vol. 11, No. 6.
- Maciel, Paulus. (2013) "Medidas das armações para surfassagem", <https://www.blogdopaulus.com/2013/06/medidas-das-armacoes-para-surfassagem.html>, Junho.
- Michaelis. (2015) "Dicionário Brasileiro da Língua Portuguesa", <https://michaelis.uol.com.br/moderno-portugues/>, Novembro.
- Plechawska-Wójcik, Małgorzata; Rykowski, Damian. (2016). "Comparison of Relational, Document and Graph Databases in the Context of the Web Application Development". In Information Systems Architecture and Technology: Proceedings of 36th International Conference on Information Systems Architecture and Technology
- Rauschmayer, Axel: "Speaking JavaScript". O'Reilly (2015). <http://speakingjs.com/es5/index.html>, Fevereiro.
- Sonoda, Rodrigo Trentin, "Ótica Prática para Consultor Ótico". 2020, Clube Autores, ISBN 978-65-00-08104-6, 2022.
- Wagner, Gerd, Diaconescu, Mircea: "Aplicativos da Web com JavaScript ou Java". De Gruyter (2017). <http://web-engineering.info/WebAppBook>,