



Biblioteca de componentes vocacionadas para a produção de GUI para pessoas com deficiência visual

ANA MARGARIDA TAVARES GUERRA

Outubro de 2021

Biblioteca de componentes vocacionadas para a produção de GUI para pessoas com deficiência visual

Ana Margarida Tavares Guerra

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em
Sistemas Gráficos e Multimédia**

Orientador: Doutor João Paulo Pereira

*Para quem partilhou o mesmo peso
Para quem não deixou de ver e sentir
Para quem vai indo e vendo, mas ficou*

Resumo

Nos dias tecnológicos em que vivemos é preciso dar atenção a todos os pormenores de usabilidades das diferentes plataformas, para que os projetos possam chegar ao maior número de pessoas e de forma igual a todas elas. Existem pessoas que não sendo totalmente invisuais padecem de algum distúrbio visual que pode condicionar o seu dia a dia e conseqüentemente o uso das diferentes tecnologias.

Considera-se o principal objetivo deste projeto a criação de uma biblioteca de componentes direcionada para a produção de *Graphical User Interfaces* (GUI) para pessoas que padeçam de algum tipo de distúrbio visual. Para solucionar o referido, foi necessário:

1. Identificar as deficiências visuais mais comuns e estudá-las para perceber quais as principais dificuldades de cada uma;
2. Realizar o levantamento das tecnologias existentes no mercado sobre as doenças selecionadas, conseguindo assim identificar as principais falhas;
3. Desenhar a melhor solução perante as informações dos pontos anteriores;
4. Desenvolver a solução segundo as boas práticas da engenharia;
5. Realizar experiências e avaliar a solução desenvolvida, validando que vai de encontro aos objetivos iniciais.

Assim o projeto visa agrupar um conjunto de distúrbios visuais e conseguir criar a biblioteca que adaptará as ferramentas aos olhos de quem os tem. Com isto as aplicações tornam-se mais inclusivas, e conseguem por isso chegar a um maior número de pessoas. O grupo de pessoas a quem se destina o projeto consegue também tirar um maior proveito das aplicações que utiliza diminuindo assim as dificuldades que podem aparecer sem o recurso desta biblioteca.

Após o estudo do estado da arte relativamente a algoritmos, aplicações e bibliotecas desenvolvidas na área, validou-se que existia apenas uma ferramenta em Flutter, uma *framework* para desenvolvimento de aplicações multiplataforma, para o caso. Como tal, e também por ser uma *framework* com utilização em crescimento contínuo, foi a escolhida para o desenvolvimento da biblioteca. Assim, foram criadas extensões e novas classes para a *framework* que contemplassem o tema visual de uma aplicação em Flutter para que fossem de fácil adaptação de um desenvolvedor a qualquer tipo de daltonismo. Foi também ainda criado o mecanismo de transformação de imagens para todos os tipos de daltonismo.

Por fim foram realizadas avaliações informais tanto por parte de desenvolvedores como por parte de pessoas com daltonismos. Ambos os pareceres foram positivos, por um lado no que diz respeito à compreensão, utilização e integração da biblioteca, e por outro na transformação das cores dos componentes e bibliotecas para diferentes tipos de daltonismo.

Palavras-chave: usabilidade, deficiência visual

Abstract

Due to the heavy use of technology in our current days, there is a special need for attention to all usability related details in every platform, so that newly developed projects can reach the highest possible number of people while maintaining the same experience. Among all the possible users, the ones with visual deficiencies can have their day-to-day activities hindered in case this is not achieved.

The main goal of this project is to create a library of components to produce Graphical User Interfaces (GUI) for people with visual impairments. To solve this problem, it was necessary to:

1. Identify the most common visual disabilities and study them to understand the main difficulties that each person has;
2. Carry out a survey of the existing technologies in the market about the selected diseases, thus being able to identify their main flaws;
3. Design the best solution based on the information from the previous points;
4. Develop the solution according to good engineering practices;
5. Experiment and evaluate the developed solution, validating that it meets the initial objectives.

Therefore, the project aims to group a set of visual disorders and to create a library that will adapt the tools to the eye sight of those who suffer these disorders. With this, the applications become more inclusive, and therefore reach a greater number of people. The group of people for whom the project is intended can also benefit more from the applications they use, hence reducing the difficulties that may arise without the use of this library.

After studying the state of the art regarding algorithms, applications and libraries developed in this area, it was discovered that there was only one tool in Flutter, a framework for developing multiplatform applications. As such, and since this is a framework with a continuously growing use, Flutter was chosen for this library development. As so, extensions and new classes were created for the framework that contemplated the visual theme of a Flutter application so that they could be easily adapted by a developer to any type of color blindness. It was also created the image transformation mechanism for all types of colour-blindness.

Finally, informal evaluations were carried out by both developers and people with colour blindness. Both opinions were positive, on the one hand regarding the understanding, use and integration of the library, and on the other hand regarding the transformation of the colours of the components and libraries for the different types of colour-blindness.

Keywords: usability, visual impairment

Agradecimentos

Primeiramente quero agradecer ao Doutor João Paulo Pereira por todo o apoio e dedicação concedidos ao longo da orientação do projeto, dando destaque à brevidade de respostas a todas as questões e dúvidas e na revisão do trabalho dando sempre as sugestões necessárias para obter a melhor qualidade possível.

Em seguida, agradecer a todos os colegas que me acompanharam ao longo deste percurso, com especial importância aos colegas que me acompanharam nesta fase final: Gil Durão, Rita Sousa e Joel Teixeira.

Sem esquecer a instituição que me acolheu, agradeço ao Instituto Superior de Engenharia do Porto, a todos os docentes e não docentes que se tenham cruzado no meu caminho e que como tal o tenham tornado melhor.

E por fim, mas não menos importante aos meus, que permitiram com que este percurso tivesse o melhor caminho e que estiveram ao meu lado apoiando sempre as minhas decisões: mãe, pai, avô, avó, Cristina, Pedro e Francisco.

Índice

1	Introdução	1
1.1	Contexto	1
1.2	Problema	2
1.3	Objetivos	3
1.4	Resultados esperados / obtidos	4
1.5	Análise de Valor	4
1.6	Abordagem Preconizada	4
1.7	Planeamento do Trabalho	5
1.8	Estrutura do documento	5
2	Contexto e Estado da Arte	7
2.1	Contexto do Problema	7
2.1.1	Tecnologias	7
2.1.2	As dificuldades de um ambliope	9
2.1.3	Doenças que afetem a visão	10
2.1.4	Cataratas	11
2.1.5	Discromatopsia	13
2.1.6	Degenerescência macular	15
2.2	Soluções/Abordagens já existentes	17
2.2.1	Cataratas e Discromatopsia	17
2.3	Tecnologias Relevantes	19
3	Análise de Valor	21
3.1	Criação de Valor	21
3.2	Valor, Valor para o Cliente e Valor Percebido	23
3.3	Proposta de Valor	24
4	Engenharia de Requisitos e Design	27
4.1	Modelo de Domínio	28
4.2	Requisitos Funcionais	29
4.3	Requisitos Não Funcionais	31
4.4	Design	32
4.5	Resumo	35
5	Implementação	37
5.1	Decisões tecnológicas	37
5.2	Flutter	39

5.2.1	Widget	39
5.2.2	ThemeData.....	42
5.2.3	<i>Plugin</i> Color_Blindness	44
5.2.4	Aplicação da transformação de cores em Imagens	44
5.3	Resumo	46
6	Experiências e Avaliação.....	47
6.1	Abordagem da avaliação	47
6.2	Experiências Realizadas	47
6.3	Resultados Obtidos.....	48
6.4	Resumo	49
7	Conclusões	51
7.1	Objetivos Alcançados.....	51
7.2	Trabalho Futuro	52
7.3	Apreciação final.....	52

Lista de Figuras

Figura 1 - Proporção de agregados familiares com acesso à internet e banda larga [1].....	2
Figura 2 - Utilização da <i>internet</i> por atividade nos últimos 3 meses antes da entrevista [1].....	8
Figura 3 - Utilização de equipamento ligados à <i>internet</i> por nível de ensino [1]	9
Figura 4 - Visão de um doente com Cataratas	13
Figura 5 - Olho humano	13
Figura 6 - Visão de um daltónico sobre lápis de cor	14
Figura 7 – Visão de um daltónico sobre semáforos	15
Figura 8 – Visão de um daltónico sobre formulários	15
Figura 9 - Visão de um doente com DM.....	17
Figura 10 - Processos da análise de valor, adaptado de [5].....	22
Figura 11 - Processo de Inovação, adaptado de [37].....	22
Figura 12 - <i>New Concept Development Model</i> (NCD), adaptado de [37]	23
Figura 13 - Proposta de Valor.....	25
Figura 14 - Modelo de domínio.....	28
Figura 15 - Diagrama de casos de uso.....	30
Figura 16 - Diagrama de Classes.....	33
Figura 17 - Diagrama de atividades Stateless Widget.....	40
Figura 18 - Diagrama de atividades Stateful Widget	41
Figura 19 - Exemplo de transformação de imagens.....	45
Figura 20 - Comparação de cores nos diferentes tipos de daltonismo	46
Figura 21 - Aplicação de teste original	48
Figura 22 - Diagrama de Classes completo	60
Figura 23 - Aplicação de testes para Tritanopia.....	61
Figura 24 - Aplicação de testes para Tritanomalia	61
Figura 25 - Aplicação de testes para Protanopia	62
Figura 26 - Aplicação de testes para Protanomalia.....	62
Figura 27 - Aplicação de testes para Deuteranopia	63
Figura 28 - Aplicação de testes para Deuteranomalia	63
Figura 29 - Aplicação de testes para Monocromacia.....	64

Lista de Tabelas

Tabela 1 – Identificação dos casos de uso	30
Tabela 2 - Descrição dos casos de uso	31
Tabela 3 - Critérios de seleção da <i>framework</i>	38
Tabela 4 - Estado final dos objetivos do projeto.....	51

Lista de Listagens

Listagem 1 - Exemplo do método Theme.of para modificar a interface	43
Listagem 2 - Exemplo de utilização do <i>plugin</i> color_blindness.....	44

Acrónimos e Símbolos

Lista de Acrónimos

AG	Atrofia Geográfica
AV	Análise de Valor
AVC	Acidente Vascular Cerebral
CPU	<i>Central Process Unit</i>
DVC	Deficiência Visual Cortical
DM	Degenerescência Macular
DVO	Deficiência Visual Ocular
EPR	Epitélio Pigmentado da Retina
FFE	<i>Fuzzy Front End</i>
GUI	<i>Graphical User Interface</i>
NCD	<i>New Concept Development</i>
NPD	Desenvolvimento de novos produtos
NPM	Node Package Manager
NVC	Neovascularização Coroide
RGB	<i>Red Green Blue</i>
UI	<i>User Interface</i>
UV	Ultra Violeta
UWP	<i>Universal Windows Platform</i>

1 Introdução

Neste capítulo são apresentados: o contexto do projeto, exibindo estudos relativos ao uso da *internet* e de aplicações em Portugal; o problema proposto, explicando a dificuldade de uso de aplicações por uma pessoa que padece de algum distúrbio visual; os objetivos que se pretendem alcançar com o desenvolvimento do projeto; os resultados esperados/obtidos; a introdução para análise de valor que será mais detalhada no capítulo 3; a abordagem selecionada para o desenvolvimento do trabalho; o planeamento do trabalho e, por fim, a estrutura que o documento irá ter.

1.1 Contexto

Este projeto é realizado no contexto da Unidade Curricular Tese/Dissertação/Estágio, do mestrado de Engenharia Informática, na área de Sistemas Gráficos e Multimédia do Instituto Superior de Engenharia do Porto e aborda temas como a usabilidade e acessibilidade das diferentes aplicações por parte de pessoas com distúrbios visuais.

Perceber o tema acima descrito, leva a que se estude anteriormente o uso de aplicações por parte de qualquer ser humano em Portugal. Esta utilização tem vindo a crescer ao longo dos tempos e a dependência de um dispositivo móvel ou de um computador é cada vez maior nas diferentes áreas do dia a dia. Segundo um estudo feito em Portugal durante o ano de 2020, 84.5% dos agregados familiares têm acesso à internet em casa e 81.7% utilizam banda larga. Em 2010, a percentagem destes valores era de 53.7% e 50.3%, respetivamente. Ambas as percentagens têm vindo a aumentar de forma muito idêntica, estando o acesso à Internet sempre acima do acesso à Banda Larga. Pode-se observar estes valores, desde o ano de 2010 ao ano de 2020 na Figura 1 [1].

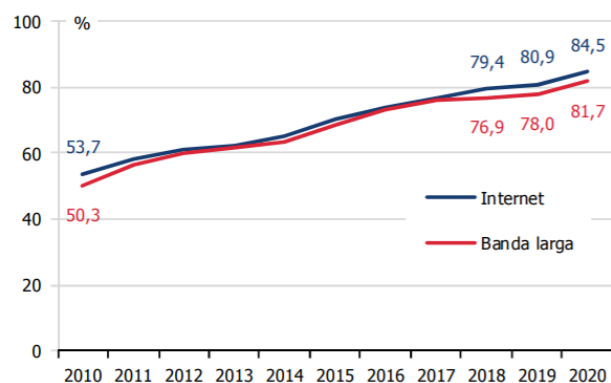


Figura 1 - Proporção de agregados familiares com acesso à internet e banda larga [1]

Com o crescimento do acesso à internet e banda larga, aumentou também o comércio eletrónico; foi em 2020 o maior aumento deste consumo a contar deste 2002. Pelo mesmo estudo indicado anteriormente, 44,5% das pessoas dos 16 aos 74 anos fizeram encomendas pela internet ao longo do ano [1].

Tanto o aumento de disponibilidade de internet pelos agregados familiares, como o aumento do consumo eletrónico, deixa claro que uma grande percentagem desta população utiliza com frequência equipamentos eletrónicos com um computador, *tablet* ou telemóvel para o efeito.

Como tal, tem existido um crescente investimento na criação de aplicações nas mais diversas áreas: saúde, organização pessoal, economia, desporto, entre outros. Este mercado revela-se assim cada vez mais exigente e minucioso, tornando de maior valor as aplicações mais atentas às necessidades de quem as vai consumir. Os consumidores tornam-se mais concentrados no desempenho, usabilidade e versatilidade da aplicação, ganhando as soluções com estas características mais pontos no mercado.

1.2 Problema

A visão é um sentido fundamental para o uso das aplicações. Existindo pessoas que não sendo totalmente invisuais, padeçam de algum distúrbio visual, a usabilidade de certas aplicações pode ficar comprometida.

Como tal, o problema que aqui é exposto remete para a usabilidade de aplicações por parte de pessoas com distúrbios visuais.

O ser humano possui outros sentidos que podem dar suporte à visão na usabilidade de aplicações e existem alguns exemplos de soluções já expostas em mercado. Estas soluções incidem principalmente na disponibilidade de áudios, como por exemplo guias de voz que descrevem as opções do menu [2], ou pela gravação de voz por parte do utilizador, como por exemplo a indicação de mudança de um canal televisivo ou a escolha de um menu numa aplicação [3].

Como é sabido, existem alternativas fáceis para conseguir adaptar a tecnologia ao grupo de pessoas que padece de algum distúrbio visual. Mas é importante para este grupo de pessoas estimular a visão e não deixar de a utilizar totalmente. Como tal, é possível adaptar as aplicações para que se possa de forma semelhante utilizar a visão na interação com esta classe de utilizadores. Com os cuidados certos e atenções no design e implementação das aplicações, o grupo de pessoas com distúrbios visuais consegue de igual forma e usando a sua visão, utilizar de forma eficaz e plena as aplicações pretendidas.

1.3 Objetivos

Exposto o contexto e o problema em causa, é definido o objetivo principal como consistindo no desenvolvimento de uma biblioteca direcionada para a criação de uma interface gráfica com o utilizador (GUI) para pessoas com algum tipo de distúrbio visual. Este projeto surge com o propósito de ajudar e simplificar o dia a dia de pessoas que se insiram no grupo à qual se destina a solução, aquando da interação com as aplicações que vão utilizar a biblioteca. Tem também como objetivos a consideração de vários distúrbios visuais na construção de apenas uma biblioteca e ainda a fácil utilização da mesma.

Para o desenvolvimento deste projeto, e para que se consiga atingir os objetivos anteriormente descritos, foram traçadas as seguintes tarefas:

1. Estudar os diferentes géneros de distúrbios visuais existentes, realizar o levantamento de quais os mais frequentes na nossa população, o que caracteriza e distingue cada um dos selecionados. Para esta primeira etapa é expectável que se realize uma intensa pesquisa bibliográfica, mas também que se realize algum contacto, através de entrevistas com entidades tais como a ACAPO – Associação dos Cegos e Amblíopes de Portugal [4]. E desta forma, conseguir reunir toda a informação necessária para a escolha e idealização de uma solução.
2. Fazer o levantamento de técnicas e tecnologias já existentes no mercado que têm como principal objetivo encontrar soluções para a usabilidade de pessoas com deficiência visual. Fazer um breve estudo sobre cada uma, (características, vantagens, desvantagens, principais queixas, ...) e reter as que se destacam no mercado.
3. Com o conhecimento adequado sobre os problemas visuais e com o estudo das tecnologias já existentes para o problema, serem criadas as principais ideias para a solução a ser desenvolvida. Desta forma começa a ser criada a abordagem mais adequada ao problema em questão.
4. Desenhar a solução escolhida, adotando sempre as boas práticas da engenharia informática, fazendo uma análise prévia do trabalho a ser desenvolvido. Realizar o levantamento de requisitos (funcionais e não funcionais), especificar os diferentes atores do projeto, descrever os casos de uso, concretizar o modelo de domínio, realizar a análise arquitetural, desenvolver os diagramas de componentes e sequência.

5. Desenvolver, de acordo com o desenho previamente efetuado e com o estudo até então realizado, a solução recorrendo às tecnologias informáticas que melhor se adequem.
6. Estabelecer um conjunto de critérios para realizar a avaliação final da solução. Esta avaliação poderá também passar pelo contacto anteriormente realizado entre entidades como a ACAPO. Seria benéfico saber a apreciação de pessoas que padeçam de algum distúrbio visual para ter uma melhor perceção sobre a usabilidade do projeto desenvolvido. Esta avaliação seguirá sempre os ensinamentos retidos ao longo do que for lecionado no módulo curricular de experimentação e avaliação.

1.4 Resultados esperados / obtidos

No fim deste projeto é expectável apresentar o estudo realizado sobre os principais distúrbios visuais na nossa sociedade, como também o levantamento das tecnologias já existentes nesta área. Será também exposta a solução final da biblioteca desenvolvida, com as expectativas de que esta esteja numa versão estável. Será também expectável que se desenvolva uma aplicação de teste para que a biblioteca possa ser testada, idealmente por pessoas que façam parte do grupo a quem se destina o projeto. Por fim, será feita também a avaliação da solução e de todo o projeto.

1.5 Análise de Valor

A Análise de Valor apresenta como principal objetivo, identificar e analisar as funções de que um projeto necessita para satisfazer as necessidades do seu cliente. Tendo em conta o aumento do valor do projeto, para os menores recursos possíveis, mas sem comprometer o resultado esperado pelo cliente e a respetiva qualidade esperada [5].

A função base e, portanto, o maior valor do projeto, é a criação de uma biblioteca de componentes de GUI que consiga demonstrar a interface igual à original para pessoas com algum distúrbio visual.

A Análise de Valor é um plano de gestão complexo e como tal é descrito com mais detalhe no capítulo 3 Análise de Valor.

1.6 Abordagem Preconizada

A abordagem adotada para a solução do problema apresentado é centralizada no desenvolvimento de uma biblioteca para criação de componentes de GUI para adaptação da usabilidade de aplicações para pessoas que padeçam de algum distúrbio visual. Esta biblioteca deve ser desenvolvida segundo as boas práticas da engenharia e prevê ajudar um conjunto de

peçoas a melhorar a usabilidade de certas aplicações no seu dia a dia. Dado que é um projeto iniciado de raiz, será estudada e analisada toda a arquitetura, escolhendo a que melhor se adequa à solução. Toda a implementação será realizada de forma a satisfazer os requisitos anteriormente levantados e para finalizar será feita a avaliação do projeto final.

1.7 Planeamento do Trabalho

Para uma melhor organização do trabalho a ser desenvolvido ao longo da dissertação, foi feito um planeamento com base no trabalho necessário, que será dividido em dois grandes grupos de tarefas.

O primeiro grupo contará com a interpretação do problema, explicação do contexto do mesmo, apresentação do estado da arte, avaliação das soluções e abordagens já existentes e ainda com uma breve análise para o design e avaliação da solução. É expectável que estes três capítulos sejam apresentados no dia 27 de fevereiro de 2021 e que antecipadamente exista uma revisão por parte do orientador Professor João Paulo Pereira.

Já o segundo grupo contará com a revisão e melhoria do grupo um, descrição pormenorizada do design, construção da solução e avaliação da mesma. Este segundo grupo é apresentado até ao dia 30 de junho de 2021, com uma revisão antecipada do orientador Professor João Paulo Pereira.

1.8 Estrutura do documento

Neste primeiro capítulo é descrito de uma forma sucinta o contexto e o problema do projeto, e é também apresentada uma breve análise de valor. São descritos os objetivos e os resultados esperados / obtidos, é também apresentada a abordagem preconizada, e o planeamento do trabalho a ser realizado.

No segundo capítulo do documento é exposto de uma forma mais detalhada o contexto do problema, são descritos o estado da arte em soluções/abordagens já existentes e o estado da arte em tecnologia relevante.

No terceiro capítulo é feita a análise de valor do projeto, definindo a criação de valor do mesmo, passando pela descrição do valor, valor para o cliente e valor percebido e por fim demonstrada e explicada a proposta de valor.

No quarto capítulo é mostrada a engenharia de requisito do projeto e o seu design também. São enumerados e explicados tanto os requisitos funcionais como os requisitos não funcionais.

No quinto capítulo é exposta a implementação da solução final, começando por explicar a escolha da tecnologia passando para a explicação propriamente dita tanto da *framework* utilizada como das alterações criadas para a criação da biblioteca.

No capítulo seis é feita a avaliação da solução final, segundo os critérios definidos e de acordo com o lecionado no módulo “Experiências e Avaliação”.

No capítulo sete são finalmente apresentadas as conclusões e as perspectivas de trabalho futuro. É feita uma retrospectiva do trabalho desenvolvido, são expostos os objetivos alcançados e não alcançados, e por fim é descrito o trabalho futuro do projeto.

2 Contexto e Estado da Arte

No capítulo do estado da arte, o contexto será exposto com mais pormenor, abordando assim estudos relativos ao uso da *internet* e aplicações por parte de pessoas com algum distúrbio visual e ainda as doenças mais comuns que comprometem a visão do ser humano. Adicionalmente será realizada uma exposição das soluções e abordagens já existentes que auxiliam os ambliopes na utilização das diferentes aplicações no seu dia a dia. Finalmente será feita uma descrição do estado da arte em tecnologias relevantes para o problema exposto.

2.1 Contexto do Problema

Como já explicado na secção 1.1, o contexto principal do problema insere-se no aumento de uso de aplicações por parte da comunidade e sobre a dificuldade que as pessoas com um distúrbio visual têm no uso dessas mesmas aplicações.

2.1.1 Tecnologias

Um estudo realizado pela Instituto Nacional de Estatística e publicado em novembro de 2020, indica que 84.5% dos agregados familiares em Portugal têm acesso à internet em casa e que 81.7% utilizam banda larga. Estes valores têm vindo a aumentar desde 2010, ano a partir do qual tem acesso ao mesmo estudo (Figura 1).

De notar que, face à pandemia COVID-19, esta percentagem de utilizadores de internet aumentou 3.0 pontos percentuais em relação ao mesmo período do ano de 2019. Nos três meses anteriores ao inquérito, uma grande parte da população utilizava a internet para comunicar ou adquirir informação, mas também para realizar atividades educativas (comunicar com os professores, assistir a aulas *online*, navegar em portais educativos e frequentar cursos *online*) ou também exercer a sua profissão em regime de teletrabalho.

Outra subida significativa no ano 2020 foi a percentagem de compras *online*, atingindo o maior aumento desde 2002, apesar de continuar abaixo da média da União Europeia (49% em 2019). Em 2020, pessoas entre 16 anos e 74 anos fizeram encomendas pela internet cerca de 44.5% nos doze meses anteriores ao inquérito e cerca de 35.2% nos três meses anteriores ao inquérito.

Este aumento de compra no comércio eletrónico pressupõe o uso dos *sites* ou aplicações disponibilizadas pelas lojas onde se destinam estas compras. Esta acessão aos *sites* ou aplicações é feita através de algum dispositivo como computador, tablet ou telemóvel.

Apesar de em 2020 a maioria da população que utilizou a internet nos três últimos meses antes do inquérito ter sido para comunicar ou aceder a informação, ouvir música é o principal motivo para a utilização da internet dentro do entretenimento com uma percentagem de 70.1%. É possível verificar na Figura 2 que as diferenças entre as percentagens de 2019 e 2020 no que diz respeito às inúmeras atividades do nosso dia a dia que requerem o uso de internet e algum tipo de dispositivo para a aceder.

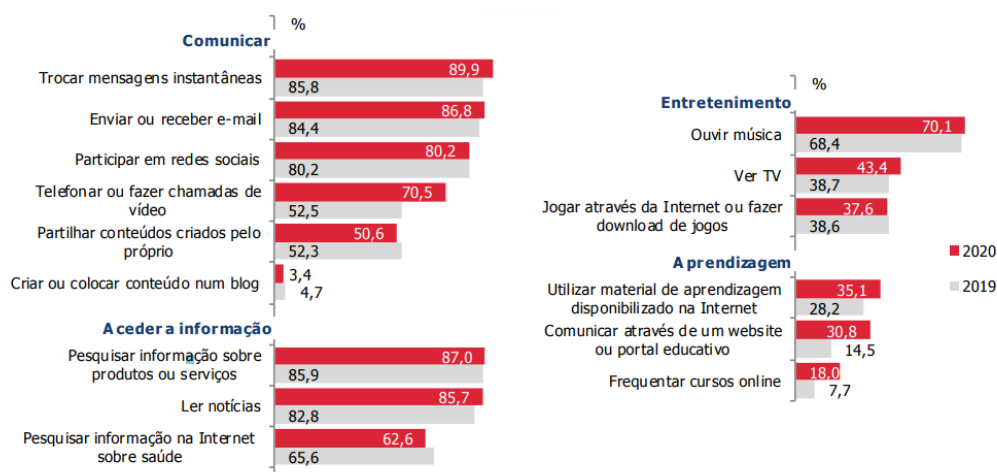


Figura 2 - Utilização da *internet* por atividade nos últimos 3 meses antes da entrevista [1]

O mesmo estudo indica que 70.5% das pessoas que utilizaram internet nos três meses anteriores à entrevista, usam algum tipo de equipamento em que o seu funcionamento se encontra ligado à internet.

Mais precisamente, como é possível verificar na Figura 3, entre os 16 anos e os 74 anos a utilização de equipamento conectado a internet varia entre 56.6% e os 81.1% consoante o nível de ensino. Este uso está ainda discriminado entre equipamento de entretenimento, equipamento de uso pessoal e equipamento doméstico [1].

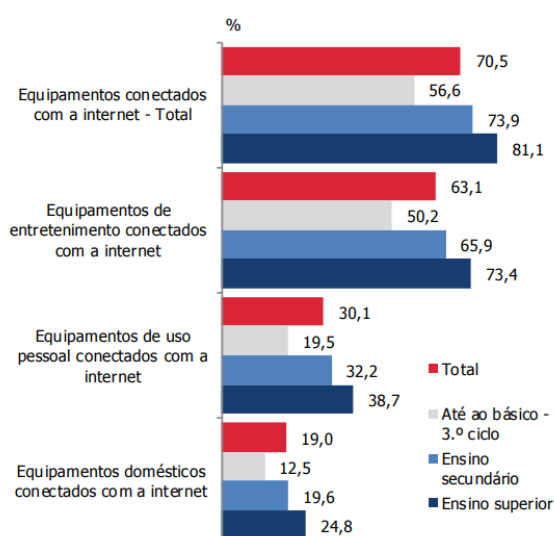


Figura 3 - Utilização de equipamento ligados à *internet* por nível de ensino [1]

Mas nem sempre esta taxa de utilização é conseguida por todas as pessoas. Estima-se que cerca de 6.5% da população mundial sofra de ambliopia¹ [6]. Segundo os censos de 2011, existem em Portugal 634 408 pessoas com deficiência (6.1%), sendo a deficiência com uma maior percentagem a deficiência visual com 1.6% [7]. Existem por isso cerca de 166 402 pessoas em Portugal que, padecendo de alguma deficiência visual, podem não conseguir usar de forma completa as tecnologias que estão disponíveis.

2.1.2 As dificuldades de um ambliope

A deficiência visual acarreta consigo limitações em praticamente todas as ações do dia a dia, como também na acessibilidade das tecnologias. Estas limitações podem levar a que exista uma restrição de usabilidade de certos equipamentos ou aplicações, que se podem facilmente ultrapassar se forem implementadas adaptações específicas para estes casos. Desta forma, torna-se assim mais abrangente o acesso à tecnologia, não deixando de parte nenhum grupo de cidadãos com deficiência.

Mas como é de conhecimento geral, nem todas as tecnologias e aplicações estão desenhadas e preparadas para as diferenças que existem entre o ser humano. Como tal, as maiores dificuldades que um ambliope apresenta ao usar tecnologia ou aplicações em qualquer plataforma são:

- Tamanho dos tipos de letra ou ícones não ajustáveis;
- Mau contraste entre fundo e texto;
- Dificuldade em aceder a teclas de atalho;
- Dificuldade em identificar de forma fácil e rápida botões de ação.

¹ Termo médico utilizado quando a visão se encontra reduzida.

Para além das dificuldades apresentadas, é de notar que apesar de em algumas situações não existir uma dificuldade em utilizar a ferramenta, existe na maioria dos casos diferença na usabilidade, ainda que não condicione a sua utilização na totalidade.

2.1.3 Doenças que afetem a visão

As doenças que afetam a visão não se baseiam apenas nos problemas a nível do olho. O sentido da visão é mais complexo, sendo que os seus problemas podem surgir no olho, na estrutura neural ou na região cerebral. Assim sendo, os diferentes distúrbios visuais podem surgir de duas deficiências diferentes [8]:

- Deficiência Visual Ocular (DVO), distúrbios que surgem diretamente no olho como por exemplo na retina, na córnea, nas células nervosas ou nos fotorreceptores (cones e bastonetes);
- Deficiência Visual Cortical (DVC), distúrbios que podem surgir tanto na estrutura neural (nervo ótico, quiasma ótico e trato ótico), estrutura responsável por levar a informação da visão até ao olho, como na região cerebral (tronco encefálico, lobo temporal e parietal e lobo occipital).

Algumas das deficiências oculares assentam sobre doenças como os Erros Refrativos (Miopia, Hipermetropia e Astigmatismo), as Cataratas, a Discromatopsia, a Retinopatia Diabética e a Degenerescência Macular.

Erros refrativos – Doença em que o olho, estando em repouso, ao receber a luz paralela, não consegue formar a imagem nítida na retina. Os três erros refrativos mais comuns são: Miopia (Dificuldade na visão ao longe, possível de corrigir com ajuda ótica adequada); Hipermetropia (Dificuldade em obter imagens nítidas de objetos a pequenas distâncias, normalmente associado a um olho demasiado pequeno, ao existir o crescimento do olho, esta doença tem tendência a diminuir) e Astigmatismo (Dificuldade em ver com nitidez a curtas e longas distâncias) [9].

Cataratas – Doença crónica normalmente associada a idade e com principais causas, genética, diabetes, tabaco, exposição aos UV. Definida como a diminuição da acuidade visual, podendo chegar a cegueira em casos mais graves. Pode ser corrigida recorrendo a cirurgia [9].

Discromatopsia – Doença que afeta os cones ou os nervos da retina e que impossibilita a distinção clara das diferentes cores e tonalidades. Ao contrário das doenças anteriores, não existe tratamento para eliminar ou atenuar esta doença [10].

Retinopatia Diabética – A diabetes é uma doença conhecida como o aumento de níveis de açúcar no sangue, que entre outras alterações pode afetar gravemente a visão. Estas alterações assentam sobre os vasos sanguíneos presentes na retina que começam por deixar sair líquido e sangue para a retina, podem também ser formados tecidos fibrosos que repuxam a retina.

Estas alteração fazem com que a visão fique comprometida podendo mesmo levar à cegueira [11].

Degenerescência macular – Doença que provoca uma lesão na mácula, zona central no fundo do humor vítreo, que permite ver claramente os pequenos detalhes. Surge com o aparecer natural do envelhecimento e apenas condiciona a visão central, não prejudicando assim a visão lateral nem a periférica do olho [12].

Por outro lado, as deficiências corticais assentam sobre doenças como a Glaucoma, Diplopia e Papiledema.

Glaucoma – Doença que afeta o nervo ótico, e que geralmente está associado a um aumento de pressão intraocular. O tratamento consiste em baixar esta pressão e pode passar por fármacos, laser ou mesmo cirurgia. Esta lesão origina a perda de visão periférica, que pode progredir para uma perda da visão central e progressivamente a cegueira [9].

Diplopia – Doença em que a imagem é estendida pelo cérebro e se passa a ver duas imagens. Esta extensão pode ocorrer na horizontal, vertical ou combinação das duas. Esta doença pode surgir, entre outras causas, pela ocorrência de doenças do sistema nervoso como esclerose múltipla, tumores cerebrais, acidentes vasculares cerebrais (AVC) e aneurismas.

Papiledema – Doença em que a visão tem tendência a escurecer conforme o tempo, por consequência de um inchaço do disco ótico, zona onde o nervo ótico intersesta o globo ocular. O diagnóstico e tratamento tardio pode levar a um atrofiamento do nervo ótico e consequências neurológicas graves.

2.1.4 Cataratas

Catarata é a denominação dada à opacidade do cristalino (estrutura biconvexa, gelatinosa e com grande elasticidade que se localiza entre a pupila e humor vítreo), doença que afeta a visão e que é considerada a maior causa de cegueira tratável atingindo 40% dos cegos.

Embora esta doença esteja relacionada com a idade, atingindo com maior incidência a faixa etária entre os 65 e os 75 anos, existem um conjunto de fatores de risco que podem provocar ou acelerar o processo. Alguns destes fatores são: medicamentos (esteroides), substâncias tóxicas (nicotina), doenças metabólicas (diabetes, doenças renais), radiações (UV, raio X), infecção durante a gravidez, entre outras.

Esta doença é normalmente diagnosticada através de um exame oftalmológico que é indicado quando o paciente se queixa de diminuição de acuidade visual, sensação de visão “nublada ou enevoada”, sensibilidade maior à luz, alteração de visão de cores ou mudança frequente da refração. Posteriormente, são realizados exames mais detalhados para que se consiga especificar a origem e o possível tratamento.

Biomicroscopia, Tonometria, Biometria, Mapeamento de retina, Topografia corneana e Ecografia são alguns dos exames realizados para que se consiga detetar a presença, mas também, perceber a extensão das opacidades cristalinas, medir a pressão intraocular, medir o comprimento axial do globo ocular, avaliar o complexo vitreoretiniano, determinar a curvatura da córnea e avaliar o segmento posterior (cavidade vítrea, retina, coroide e nervo ótico). Com a análise destas especificações, o possível tratamento da doença passa pela cirurgia, denominada de Facectomia, que consiste na substituição do cristalino opaco por uma prótese designada de lente intraocular [13].

Esta doença é caracterizada por períodos de evolução, que podem condicionar a cirurgia, como:

Incipiente – referente ao desenvolvimento da opacidade que inicia na periferia e progride até ao centro do cristalino;

Intumescente – referente ao momento em que o cristalino incha devido à retenção de água, fazendo com que a íris seja “empurrada”, o que diminui a largura da câmara anterior (escurece a visão);

Maduro – referente à perda do excesso de líquido do cristalino o que faz com que este se retraia, formando assim opacidades com tons castanhos e cinzentos;

Hiper maduro – referente à circunstância em que o cristalino perde as suas linhas irradiadas e se torna todo homogéneo, perdendo também fluido e aumentando a profundidade da câmara anterior.

Relativamente à classificação, esta pode advir segundo:

- Localização da opacidade (catarata subcapsular; catarata nuclear; catarata cortical; catarata “árvore de Natal”);
- Grau de maturidade (catarata imatura; catarata em evolução; catarata hiper madura; catarata morganiana);
- Idade de início (congénita, infantil, juvenil, presenile, senil);
- Doenças sistémicas (diabetes, dermatite atópica, distrofia miotónica, neurofibromatose);
- Catarata secundária;
- Catara tóxica.

Como podemos verificar na Figura 4, existem diferenças entre a visão de uma pessoa sem Cataratas (lado esquerdo da imagem) e a visão de uma pessoa com cataratas (lado direito da imagem). A imagem vista pelas pessoas que possuem a doença torna-se mais escura e baça consoante o nível da mesma. Esta diferença leva a que exista uma perda de acuidade e percepção na visão do dia-a-dia, bem como na experiência de utilização das diferentes tecnologias.



Figura 4 - Visão de um doente com Cataratas

2.1.5 Discromatopsia

A discromatopsia, mais conhecida como daltonismo, é uma perturbação na visão das cores que afeta cerca de 200 milhões de pessoas em todo o mundo [10]. É uma incapacidade ou diminuição de capacidade de ver a cor ou de distinguir as diferenças entre as várias cores. A discromatopsia é vulgarmente chamada de daltonismo devido ao primeiro cientista a estudar a doença. Assim foi homenageado o químico John Dalton, que estudou a anomalia de que ele próprio era portador.

Na maioria dos casos, o daltonismo é uma doença maioritariamente genética, que pode ser hereditária, recebida pelos pais ou avós. Mas também pode ser adquirida, por alguma lesão na parte estrutural dos olhos ou em alguma célula nervosa da retina. Para a primeira possibilidade a doença consiste numa ausência ou numa mutação, de um ou mais cones. Já a segunda possibilidade acontece quando existe algum problema nas células nervosas que se encontram na área da retina (Figura 5).

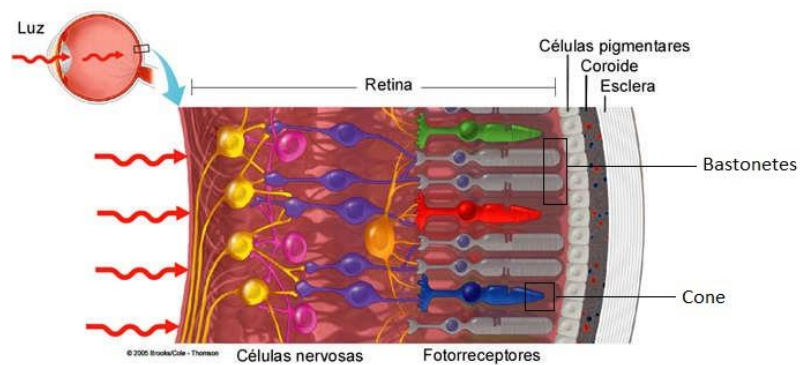


Figura 5 - Olho humano

Esta doença é muitas vezes diagnosticada na altura escolar uma vez que são realizados trabalhos manuais que pressupõem a distinção clara entre as diferentes cores, no entanto existem daltónicos que só se apercebem desta dificuldade na fase adulta.

Esta perturbação está diretamente relacionada com o cromossoma X, uma vez que os genes que produzem fotopigmentos estão presentes neste cromossoma. É por surgir no cromossoma

X que esta doença atinge maioritariamente os homens, uma vez que para a doença estar presente nas mulheres teria de se verificar nos dois cromossomas X. Estima-se que esta doença afete cerca de 6% a 10% o sexo masculino e entre 0.4% e 0.7% o sexo feminino [14].

O daltonismo não se avalia por níveis de dificuldade na visão, mas sim por três tipos principais, sendo estes:

Monocromacia – Problema visual em que as pessoas apenas conseguem distinguir as cores no tom cinza, passando assim a ver tudo nessas tonalidades. Este problema apresenta-se quando existe anomalia ou ausência dos três cones;

Dicromacia – Acontece quando existe uma ausência de um dos cones. Sendo por isso o problema visual em que as pessoas não conseguem distinguir as tonalidades referentes ao cone em falta. A Dicromacia está dividida em três tipos: Protanopia (Ausência na retina de cones vermelhos), Deuteranopia (Ausência na retina de cones verdes) e Tritanopia (Ausência na retina de cones azuis).

Tricomacia – Acontece-se quando existe uma anomalia num dos cones. Sendo por isso o problema visual em que as pessoas distinguem com alguma dificuldade as tonalidades referentes ao cone afetado. A Tricomacia está dividida em três tipos: Protanomalia (Presença de uma irregularidade na célula pigmentar associada aos cones vermelhos), Deuteranomalia (Presença de uma irregularidade na célula pigmentar associada aos cones verdes) e Tritanomalia (Presença de uma irregularidade na célula pigmentar associada aos cones azuis).

Como se consegue perceber, o dia a dia de uma pessoa daltónica fica completamente diferente ao de uma pessoa que consegue ver todos os espectros de cores, e algumas situações podem até mesmo ser difíceis de ultrapassar até que haja um diagnóstico e adaptação à doença.

Como é verificado na Figura 6, existe uma grande diferença no espectro de cores visíveis para um daltónico, por exemplo num conjunto de lápis de cor, onde estes são usados com frequência no ensino preparatório e primeiro ciclo. É nesta fase do desenvolvimento do ser humano que são diagnosticados a maioria dos casos de daltonismo.



Figura 6 - Visão de um daltónico sobre lápis de cor

Já numa fase adulta e com um maior risco de se tornar uma situação complicada caso não exista um diagnóstico e uma sabedoria para lidar com a doença a questão da visão na condução de um veículo pode tornar-se perigosa. A Figura 7 mostra a diferença criada por três tipos diferentes de daltonismo na visão de um semáforo.



Figura 7 – Visão de um daltónico sobre semáforos

Estas dificuldades aquando do uso da tecnologia também se fazem sentir e é esse assunto que se irá tentar minimizar ao máximo com este projeto. O exemplo que é apresentado é o preenchimento de formulários que, como se pode verificar na Figura 8, é fácil para uma pessoa não daltónica perceber se os campos preenchidos estão corretos ou incorretos, através do rebordo verde ou vermelho respetivamente, sobre os campos preenchidos. Para uma pessoa com algum tipo de daltonismo esta perceção de correto ou incorreto pode não ser tão clara e provavelmente só após a tentativa de submissão se aperceberá que algum campo não está correto.

The image shows two side-by-side screenshots of a web form. The form has the following fields: 'Name' (input with 'Bob'), 'Surname (optional)' (empty), 'Age' (empty), and 'Remember me' (checkbox). Below the fields are two 'Save' buttons. In the left screenshot, the 'Name' input has a green border, and the 'Age' input has a red border. In the right screenshot, the 'Name' input has a yellow border, and the 'Age' input has a yellow border. The 'Remember me' checkbox is checked in both.

Figura 8 – Visão de um daltónico sobre formulários

2.1.6 Degenerescência macular

A degenerescência macular (DM) é uma doença indolor, crónica e progressiva relacionada com o envelhecimento e responsável por uma parte significativa de perda de visão irreversível em idades avançadas.

Esta degenerescência é assinalada com a deterioração dos fotorreceptores e epitélio pigmentado da retina (EPR) (células pigmentares na Figura 5) onde são criados depósitos extracelulares anormais intitulados de drusa.

Os sintomas mais evidentes conforme o avanço da doença são: a perda gradual da visão no seu campo central, mantendo sempre intacta a visão periférica; criação de drusas, alterações nos pigmentos EPR, Neovascularização coroide (NVC) (criação e novos vasos sanguíneos na Coroide, Figura 5).

Considera-se que é uma doença que pode resultar da interação de fatores individuais e ambientais que leva ao aumento do risco de desenvolver a mesma. Por outro lado, o conhecimento sobre estes fatores leva a uma possível prevenção e ao início das terapias que atrasam o seu desenvolvimento e progressão. Alguns destes fatores de risco são: a idade (considerado o fator mais importante e consistente); tabagismo e suplementação dietética (fatores de risco modificáveis); fatores genéticos; doenças cardiovasculares e cor da íris (fatores ainda em estudo). [15]

Esta patologia pode ser caracterizada das seguintes formas: DM não exsudativa e DM exsudativa ou, segundo a sua progressão, como DM precoce, DM intermédia ou DM avançada. Esta classificação tem implicações a nível de tratamento e prognóstico da doença.

DM não exsudativa – É a forma mais comum da doença com cerca de 90% dos casos diagnosticados, e é caracterizada pela criação de drusas amareladas ao nível do EPR, que implicam modificações pigmentares e acontece normalmente de modo progressivo. Em etapas avançadas a perda da visão pode ser irreversível pela junção das áreas afetadas, processo designado de atrofia geográfica (AG).

DM exsudativa – Representa apenas 10% dos casos e é caracterizada pela formação de proliferação de vasos anómalos na mácula que resultam na acumulação de fluído e risco de hemorragias. Ao contrário da anterior podem ocorrer perdas visuais súbitas e profundas.

DM precoce – Diagnosticada quando existem muitas drusas pequenas (<63 µm de diâmetro) ou algumas drusas intermédias (63 a 124 µm de diâmetro), ou alterações pigmentares.

DM intermédia – Diagnosticada quando existem muitas drusas intermédias ou pelo menos uma drusa grande (> = 125 µm), ou AG que não envolve o centro da fóvea (mancha amarelada centrada na retina onde se concentram os cones e onde é criada a imagem enviada para o cérebro).

DM avançada – Diagnosticada quando existe AG que envolve o centro da fóvea ou qualquer evidência de NVC.

Embora não cause cegueira total, a perda progressiva e irreversível da visão central acarreta grandes dificuldades no que diz respeito às diferentes tarefas do quotidiano. Na Figura 9 podemos ver a diferença entre a visão de uma pessoa sem DM e com DM, onde verificamos a perda progressiva do centro da visão.



Figura 9 - Visão de um doente com DM

2.2 Soluções/Abordagens já existentes

As tecnologias usam de forma intensa variadíssimas cores, estas são essenciais no design de interface. São utilizadas, para a nível estética, tornar as plataformas mais atrativas, mas também porque atualmente têm um papel importante no que diz respeito a distinção e compreensão da informação que se quer transmitir.

No entanto, esta transmissão de informação através das cores ou até a apreciação da estética das plataformas é condicionada quando vista por pessoas que padeçam de daltonismo, cataratas ou degenerescência macular.

Com o intuito de ajudar no desenvolvimento de conteúdo para daltónicos o W3C estabeleceu um conjunto de diretrizes que auxiliam as empresas na seleção de cores mais indicadas para esta doença [16]. No que diz respeito à progressiva perda de visão pelas pessoas que padeçam da doença cataratas ou degenerescência macular existem pequenas correções de imagem que se podem aplicar para que sejam de mais fácil interpretação para estas pessoas.

Neste sentido o problema tem sido estudado e surgiram algumas alternativas.

2.2.1 Cataratas e Discromatopsia

Uma vez que a catarata é caracterizada como a doença da opacidade do cristalino, ao conseguir aumentar o contraste da imagem, esta torna se mais nítida ao utilizador que têm a doença. Por outro lado, para tornar a imagem de mais fácil interpretação a um doente com daltonismo é necessário realizar uma seleção de cores adequada.

Com estes estudos surgiram algoritmos que auxiliam os designers na seleção de paleta de cores (1), algoritmos para ajustar o contraste das diferentes cores (2), algoritmos que ajustam as cores a uma determinada paleta (3) e ainda algoritmos que preservam a matriz de cores e fazem a conversão das cores mais delicadas ao olho do daltónico para cores o mais próximo destas possível e acessíveis a este grupo de pessoas (4 e 5) [17].

1. Troiano et al. apresentam um algoritmo genético para a seleção de uma paleta de cores que sejam distinguíveis pelos daltônicos. Estas cores são selecionadas conforme os requisitos estéticos e os requisitos de acessibilidade [18];
2. Mereuta et al. apresentaram um algoritmo para a alterações de contraste de cores utilizando otimização numérica. Foram testados vários conjuntos de parâmetros para testar a rapidez de cada um [19];
3. Jefferson et al. apresentam um algoritmo que mapeia as cores usando os critérios de avaliação W3C. Este algoritmo começa por selecionar o conjunto de cores da imagem que é necessário corrigir, calcula a distância de cores usando os critérios de avaliação da W3C e calcula as cores necessárias a alterar [20];
4. Jian-Bin et al. apresentam um algoritmo que remapeia os componentes da matriz no espaço de cores HSV com base nas características das imagens. Tem como objetivos: manter a iluminação e a saturação para conseguir manter a imagem natural, preservar a ordem da matriz e um cálculo rápido [21];
5. Kuhn et al. apresentam um algoritmo eficiente para a transformação de cores em imagens. Este algoritmo tem como principais distinções dos anteriores a preservação, tanto quanto possível, das cores originais da imagem e a rapidez com quem é feita a transformação das cores. Este algoritmo foi testado e avaliado pela positiva por um grupo de pessoas daltônicas. Adicionalmente existe também uma extensão para alteração do contrastes das cores [22].

Para além dos algoritmos criados ao longo do tempo de forma a conseguirem atenuar a dificuldade do daltónico na visão das cores, foram também criadas aplicações cujos seus objetivos são: identificação do tipo de daltonismo; auxiliar a identificação de cores; transformas as cores para que estas possam ser distinguidas entre a comunidade de pessoas daltónicas e ainda simulação da visão de cores por um daltónico. São exemplo destas aplicações a *SeeColors*, a *Color Binoculars* e a *Color Blind Pal*.

SeeColors – Aplicação criada pela Samsung em parceria com a Colorlite em 2017 com o intuito de ajudar os daltónicos a perceberem o tipo de daltonismo que têm. Esta aplicação identifica o tipo de daltonismo com a ajuda de testes realizados pelo utilizador e após o diagnóstico adapta as cores produzidas quer em televisão quer em dispositivos móveis consoante os resultados dos testes [23].

Color Binoculars – Aplicação criada pela Microsoft em 2016 com o principal objetivo de ajudar os daltónicos a distinguir as cores no seu quotidiano. Ao utilizar a aplicação e consequentemente aceder à câmara do dispositivo móvel, a aplicação altera as combinações de cores que sejam mais difíceis de distinguir ao utilizador. No caso a aplicação apenas está preparada para alterar as combinações de cores vermelho/verde e azul/amarelo. No primeiro

caso os tons vermelhos são tornados mais claros e brilhantes e os tons verdes são tornados mais escuros e saturados. Esta aplicação apenas está disponível para iOS [24].

Color Blind Pal – Aplicação criada por daltónico e para daltónico em 2015. Com o principal objetivo semelhante à *Color Binoculars*, esta aplicação consiste na transformação das cores que são captadas através da camera do dispositivo móvel. Ao contrário da aplicação *Color Binoculars* esta aplicação não altera duas combinações de cores, está sim preparada para todo o espectro de cores disponível. Para além das alterações de cores, a aplicação também disponibiliza a descrição de cada cor presente na captação da imagem bem com o código RGB. Esta aplicação está disponível para Android e iOS [25].

Após a chegada das diferentes aplicações para reconhecimento de cores e alterações de contrastes e com o aumento deste tema no mercado, também a Microsoft [26], a Google [27] e a Apple [28] sentiram a necessidade de adaptar os seus dispositivos para que estes estejam preparados a ser utilizados por pessoas com daltonismo e cataratas. Neste sentido, tantos os dispositivos Android com os dispositivos iOS têm a funcionalidade de adaptar as suas cores aos diferentes tipos de daltonismo, tal como a última versão Windows.

No caso dos dispositivos Android o sistema oferece ao utilizador a possibilidade de ajustar a correção de cor aos tipos Deuteranomia, Protanomia e Tritanomia [29].

Na última versão do sistema operativo idealizado pela Microsoft, Windows 10, e nos dispositivos iOS o sistema oferece a possibilidade de ajustar a correção de cor aos tipos: Cinzentos, Protanopia, Deuteranopia e Tritanopia.[30] Nos dispositivos iOS é ainda possível ajustar a intensidade e a tonalidade para cada uma das opções anteriores [28].

2.3 Tecnologias Relevantes

Para além das tecnologias apresentadas no subcapítulo 2.2 existem também já no mercado um conjunto de bibliotecas e *plugins* relativas ao daltonismo. Os apresentados neste subcapítulo passam por: simulações de daltonismo para que os desenvolvedores possam ir validando aos olhos de um daltónico a estética dos seus projetos e alterações de esquemas de cores.

Relativamente a biblioteca de JavaScript, foram encontrados no Node Package Manager (NPM) sete resultados sendo que a biblioteca com maior percentagem de popularidade é a biblioteca *color-blind* com 9% e a com maior qualidade a biblioteca *bw-color-blind-palette* com 76% [31].

Também a linguagem R tem uma biblioteca que simula a visão de um daltónico perante o resultado em gráficos. Ao ser usada esta biblioteca desenvolvedor consegue perceber com é que uma pessoa daltónica vai observar os resultados obtidos através dos diferentes gráficos [32].

RGBInd é uma biblioteca de uma ferramenta de simulação que têm como objetivo ajudar os desenvolvedores de web como os utilizadores daltónicos vão visualizar os seus projetos e ajuda ainda na seleção de paletas de cores para uma maior legibilidade. Está disponível apenas para a protanopia e a deuteranopia [33].

ColorBlindnessFlutter é uma biblioteca desenvolvida para Flutter que surge em 2020 como a primeira biblioteca desenvolvida sobre o tema daltonismo. A biblioteca permite alterar a paleta de cores para que seja possível suportar os tipos de daltonismo de Dicromacia e Tricomacia [34].

Foram também encontradas duas extensões para o browser Google Chrome que auxiliam os daltónicos na visão das cores neste browser em específico, são elas:

Colorblind – com última atualização em 2018 e adaptado para Tritanopia, Deuteranopia e Protanopia, esta extensão permite que pessoas daltónicas consigam ver mais cores para além daqueles que tipicamente têm dificuldade [35];

Let's get color blind – com última versão em 2020, simula a informação recebida por um daltónico e faz também a transformação das cores necessárias usando filtros para melhorar as cores quando vistas por daltónicos [36].

3 Análise de Valor

Identifica-se a Análise de Valor (VA) como um processo de desenvolvimento que visa satisfazer as necessidades do utilizador, identificando as partes do projeto onde se pode aumentar o seu valor. Esta análise e melhoria deve optar por soluções que forneçam mais valor ao produto, com o menor custo possível, mas sem comprometer a qualidade do mesmo. Para descrever a VA do projeto, decidiu-se dividir este capítulo em três partes:

- Criação de Valor, onde será descrito o processo da análise de valor, modelo de *Fuzzy Front End* e o *New Concept Development Model*. Serão ainda identificadas e analisadas as oportunidades do projeto;
- Será descrito o Valor, Valor para o cliente e Valor percebido fazendo a ligação com o projeto;
- Será apresentada a proposta de valor do projeto.

3.1 Criação de Valor

O processo da Análise de Valor é composto pelos seguintes cinco processos (Figura 10):

1. Orientação – Formar a equipa da VA e a realizar a seleção do produto;
2. Análise Funcional – Definir e estruturar as funcionalidades do projeto;
3. Criação de alternativas – Identificar e debater as alternativas para o nosso projeto;
4. Análise e avaliação – Analisar e avaliar das alternativas selecionadas;
5. Implementação – Implementar o projeto.

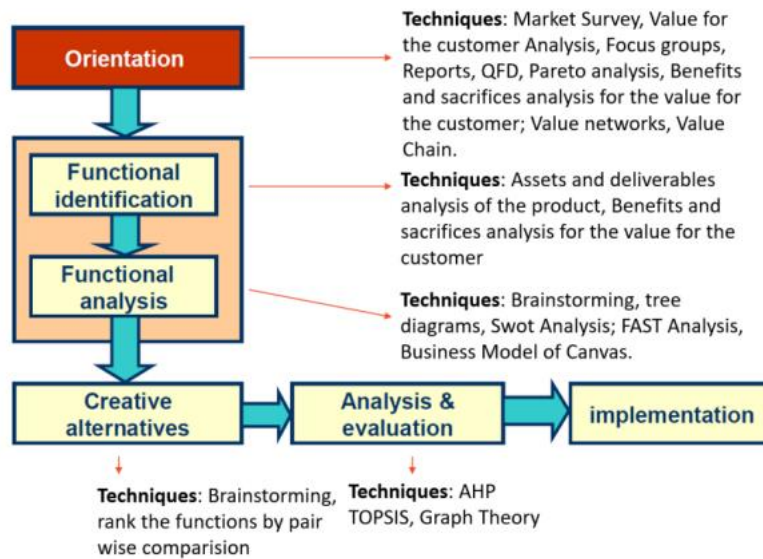


Figura 10 - Processos da análise de valor, adaptado de [5]

Para a primeira fase, e no que diz respeito ao processo criativo, é proposto um processo de inovação que agrega três fases: *Fuzzy Front End* (FFE), Processo de Desenvolvimento de Novos Produtos (NPD) e Comercialização, conforme identificado na Figura 11.

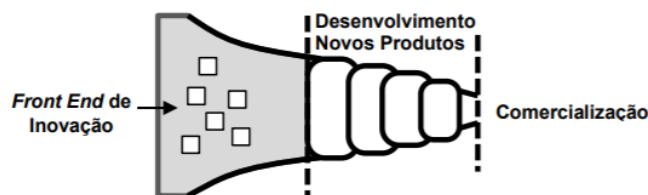


Figura 11 - Processo de Inovação, adaptado de [37]

Apesar da primeira etapa ser considerada a que apresenta mais oportunidades de melhoria da inovação, ao longo do tempo, foram identificadas algumas lacunas na sua utilização. Sendo a principal lacuna a dificuldade de comparar as práticas de FFE entre as empresas, por não existirem termos e definições comuns para certos elementos chave. Foi assim criado o *New Concept Development Model* (NCD) que surge com três partes principais (Figura 12):

1. Motor – Identifica-se com sendo a liderança e o impulsionador dos cinco elementos chave que são controlados entre si pela sua cooperação;
2. Área Interna – Onde são apresentados os cinco elementos chave (Identificação de Oportunidades; Análise de Oportunidades; Geração e Enriquecimento de Ideias; Seleção de Ideias e Definição do Conceito);
3. Fatores de Influência – Capacidades organizacionais; Mundo externo (ex. legislação, política, economia, cultura).

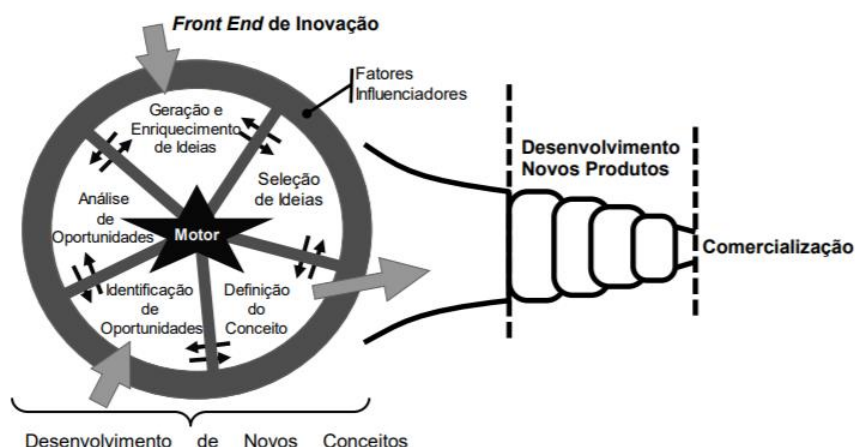


Figura 12 - *New Concept Development Model (NCD)*, adaptado de [37]

Este modelo apresenta-se de forma circular para dar a ideia de que os cinco elementos devem fluir sobre si e devem manter-se interligados [37].

Relativamente à identificação de oportunidades e enquadramento modelo NCD neste projeto, são identificadas como principais oportunidades, a criação de uma biblioteca de adaptação de componente GUI para pessoas que padeçam de distúrbios visuais. Esta biblioteca distinguir-se-á das já existentes no sentido de abranger vários tipos de distúrbios e conseguir abranger numa só biblioteca várias alterações que se adaptaram ao tipo de distúrbio que o utilizador tenha. Será também uma oportunidade, a adaptação individual do utilizador ao tipo de distúrbio que este padeça, e não ter apenas uma solução genérica como são a maioria das soluções existentes. Temos como exemplo o daltonismo que apresenta vários tipos e que a maior parte das soluções já existentes apenas estão preparadas para tratar alguns tipos do daltonismo.

Fazendo uma análise das oportunidades anteriormente definidas, é de notar que reunir mais do que uma deficiência visual na mesma biblioteca é interessante no ponto de vista em que o desenvolvedor acaba por englobar distúrbios que poderiam ainda não estar no plano, um maior número de doenças é abordado e conseqüentemente mais pessoas conseguem usufruir das plataformas onde a biblioteca se aplique. E ainda o facto de não ser necessário utilizar uma biblioteca por cada doença visual a que se queira chegar e por isso poupa trabalho no sentido de estar tudo concentrado apenas numa biblioteca. No que diz respeito à adaptação individualizada de cada distúrbio, esta oportunidade faz com que as pessoas que padeçam de distúrbios visuais não seja todas generalizadas para os distúrbios com maior número de pessoas.

3.2 Valor, Valor para o Cliente e Valor Percebido

Valor, é uma palavra cujo significado depende do cliente e do contexto onde está a ser usada, esta é uma definição importante para conseguir tornar claro as mais valias e as prioridades do projeto. Embora possam existir inúmeras definições de valor, existem características comuns a

todas elas, são elas: alto nível de desempenho, capacidade, impacto emocional e estilo em relação ao seu custo. O valor pode assim ser expresso através da equação:

$$Valor = \frac{(desempenho+capacidade)}{custo} = \frac{Função}{Custo}.$$

Através da análise da equação percebe-se que o valor aumenta, quando existe um aumento da função (desempenho e capacidade) em maior escala que o seu custo. Quanto maior a diferença entre estas duas variáveis, maior se torna o valor [5].

Já de um ponto de vista do cliente, valor é, segundo Tony Woodall, qualquer percepção do cliente que resulta na sua ligação com o produto oferecido por uma organização, podendo ter origem numa redução do esforço, na presença de benefício ou até mesmo na combinação de ambos [38]. “Clientes diferentes percebem valores diferentes para os mesmos produtos ou serviços.” [5].

Outra definição importante para a análise de valor de um projeto é o conceito de valor percebido, diferenciado do valor para o cliente.

“Lindgreen e Wynstra afirmam que o valor, conforme percebido pelo produtor, significa algo diferente do valor percebido pelo cliente, ou seja: o produtor é menos sensível ao preço, enquanto o consumidor é mais sensível na qualidade do produto.” [5]

“Zeithmal sugeriu que o valor percebido é a avaliação geral do consumidor da utilidade de um produto com base nas percepções do que é recebido e do que é dado”. [5]

Indica-se como valor do projeto a elaboração de uma biblioteca de componentes de GUI destinada a pessoas que padeçam de algum distúrbio visual. Neste sentido, existe uma maior acessibilidade deste grupo de pessoas a diferentes plataformas.

Indica-se como valor para o cliente a ajuda que este vai ter na utilização das plataformas que utilizem a biblioteca desenvolvida. O cliente passará assim a ter menor dificuldade na utilização destas ferramentas e conseguirá de forma mais plena e clara usufruir destas.

Indica-se como valor percebido o valor idêntico aos dois anteriores conjugados para o grupo de pessoas que padeça de algum distúrbio visual. Considera-se também como valor percebido o facto das pessoas que não padeçam de algum distúrbio visual conseguirem ter a percepção da visão daqueles que padeçam de algum distúrbio visual.

3.3 Proposta de Valor

A proposta de valor é usada tanto a nível profissional como a nível académico e tem origem no trabalho de Lanning e Michael em 1988 que se divide em duas partes. Por um lado, o desenvolvimento da proposta de valor, e por outro a criação do sistema de entrega de valor.

Numa proposta de valor está presente a definição do que a organização pretende oferecer ao cliente, diz respeito ao cliente, mas para uso interno da empresa. Deve também estar definido como a organização trabalha, indicando as atividades que exercem para fornecer o melhor serviço, mas mantendo o lucro da organização. [39]

Desta forma é apresentada a proposta de valor através da Figura 13 onde é possível ver o lado do perfil do cliente e o lado da proposta de valor. Para o primeiro caso são apresentadas as tarefas do cliente, os seus ganhos e as suas dores e para o segundo caso são apresentados os produtos e serviço, os criadores de ganhos e os alívios para as dores.

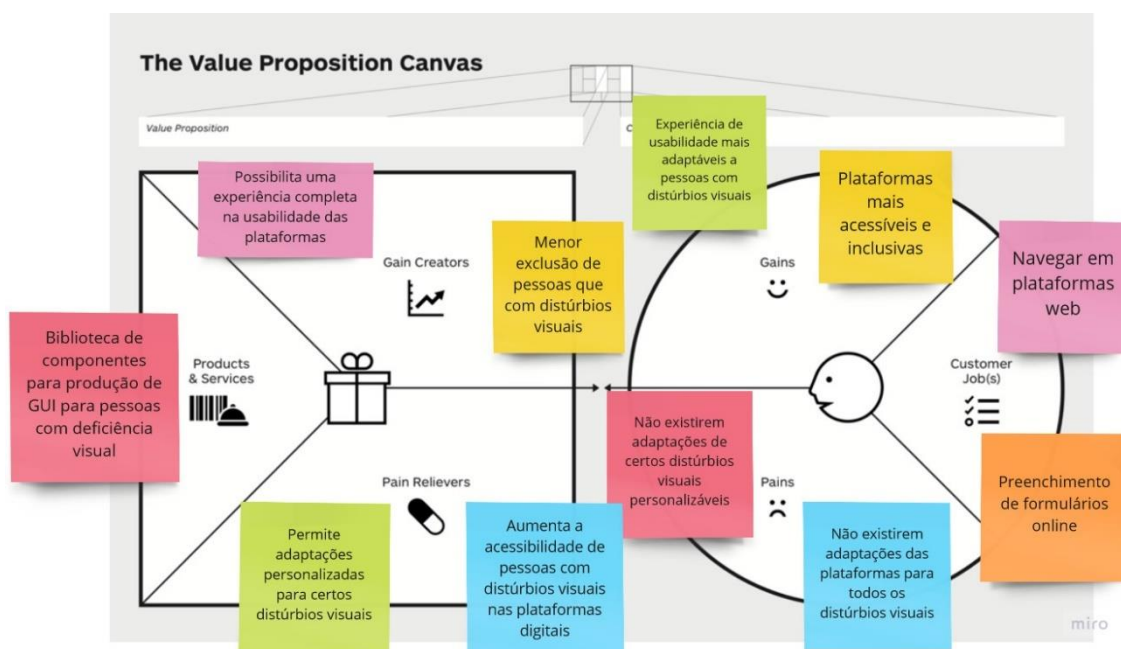


Figura 13 - Proposta de Valor

Para o perfil do cliente destaca-se:

Customer Jobs – Preenchimento de formulários online; Navegação em plataformas web;

Gains – Plataformas mais acessíveis e inclusivas; Experiência de usabilidade mais adaptáveis a pessoas com distúrbios visuais;

Pains – Não existirem adaptações de certos distúrbios visuais personalizáveis; Não existirem adaptações das plataformas para todos os distúrbios visuais.

Relativamente à preposição de valor destaca-se:

Products & Services – Biblioteca de componentes para a produção de GUI para pessoas com deficiência visual;

Gain Creators – Possibilita uma experiência completa na usabilidade das plataformas; Menor exclusão de pessoas que padeçam de distúrbios visuais;

Pain Relievers – Permite adaptações personalizadas para certos distúrbios visuais; aumenta a acessibilidade de pessoas com distúrbios visuais nas plataformas digitais.

4 Engenharia de Requisitos e Design

No capítulo destinado à Engenharia de Requisitos e Design, é apresentado o modelo de domínio, são descritos os requisitos funcionais (relativos ao aspeto funcional do software, requisitos específicos do cliente) e não funcionais inerentes ao projeto com o auxílio do modelo FURPS+ [40] e é apresentada e explicada a arquitetura pensada para o desenvolvimento do projeto adotando as boas práticas da engenharia de software. O modelo FURPS+ classifica a qualidade dos requisitos de software segundo as seguintes categorias [40]:

- Funcionalidade – relativa a funcionalidades transversais a diferentes projetos e negócios que não específicos do domínio ou negócio em causa, Como por exemplo: autenticação, autorização, auditoria, localização, envios de emails e relatórios;
- Usabilidade – relativa à interface com o utilizador (prevenção de erros, estética e design, ajudas);
- Confiabilidade – relativa à integridade, conformidade e interoperabilidade do software;
- Desempenho – relativo ao desempenho do software, como é exemplo o tempo de resposta, consumo de memória e utilização da CPU;
- Suportabilidade – relativa aos requisitos de suportabilidade, nomeadamente, testabilidade, adaptabilidade, manutenibilidade, compatibilidade e configurabilidade;

O símbolo + agrega outros requisitos não funcionais tais como por exemplo:

- Desenho – relativo às restrições de design do sistema (linguagens de programação; ferramentas de desenvolvimento; bibliotecas; etc.);
- Implementação – relativo às restrições e especificações da construção do sistema (padrões obrigatórios; linguagens de implementação; políticas de dados; limites de recursos; etc.);
- Interface – relativo às restrições e especificações das funcionalidades intrínsecas do sistema com o utilizador;

- Físicos – relativo às restrições e especificações das limitações físicas pelo hardware necessário.

4.1 Modelo de Domínio

Na Figura 14 é apresentado o modelo de domínio com os conceitos de negócio da biblioteca desenvolvida. Assim, considera-se o projeto desenvolvido o conceito “Biblioteca” que pode ser utilizada no desenvolvimento de uma ou mais aplicações. Estas por sua vez podem ser desenvolvidas por vários desenvolvedores. Consecutivamente, as aplicações desenvolvidas são utilizadas por utilizadores. Estes utilizadores podem padecer de várias deficiências visuais, como é o caso da Degenerescência Macular, Discromatopsia ou Cataratas.

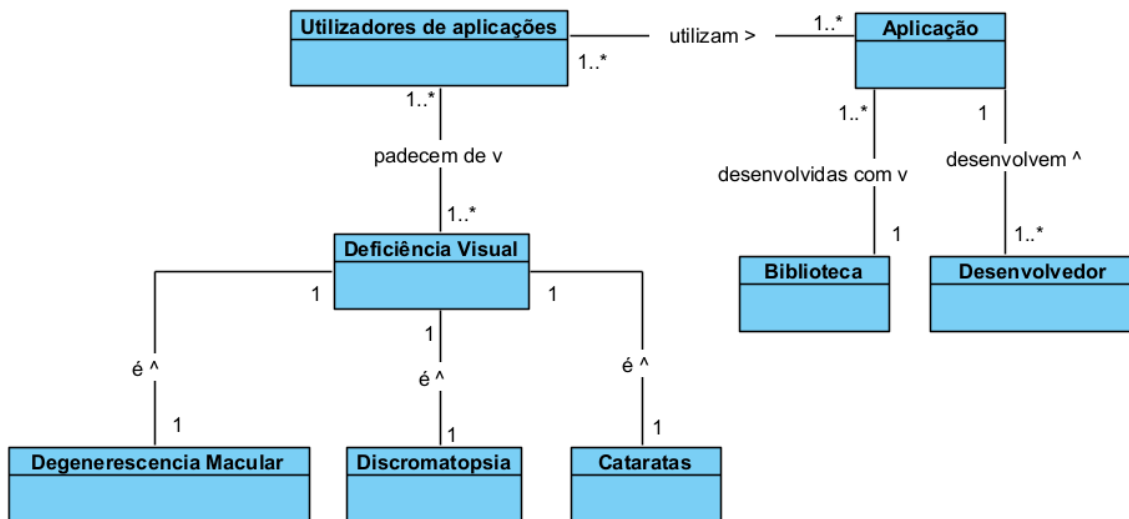


Figura 14 - Modelo de domínio

4.2 Requisitos Funcionais

São considerados os requisitos funcionais todos aqueles que têm uma ligação direta entre as funcionalidades de interação dos utilizadores e do software. Após a identificação dos atores do projeto são enumeradas as funcionalidades que cada ator espera obter da biblioteca.

Apesar de o principal utilizador da biblioteca ser o desenvolvedor das aplicações que pretende ajustar o seu projeto a pessoas com deficiências visuais, os atores mais importantes e que vão usufruir de forma indireta com a biblioteca, são as pessoas que padecem de algum distúrbio visual. Assim foram identificados como atores do projeto, ainda que de uma forma indireta, as:

1. Pessoas sem qualquer deficiência visual;
2. Pessoas que padeçam de discromatopsia (qualquer um dos sete tipos);
3. Pessoas que padeçam da doença Cataratas;
4. Pessoas que padeçam da doença DM.

Assim, para as pessoas que não padeçam de nenhum distúrbio visual, é expectável que vejam a aplicação tal como foi desenhada.

Já para as pessoas que padeçam de daltonismo é esperado que a biblioteca modifique as cores presentes na aplicação para a paleta de cores que o daltónico consegue distinguir. Tem em consideração cada tipo de daltonismo explicado na secção 2.1.5.

Por outro lado, para as pessoas que padeçam da doença Cataratas, é previsto que a biblioteca consiga adaptar as tonalidades das cores e a sua intensidade para que estas sejam de visão mais clara.

Por fim, para as pessoas que padeçam da doença DM, é expectável que a biblioteca consiga adaptar o centro de imagens para que a mancha negra da visão destes doentes seja atenuada.

As perspetivas de cada ator e a descrição de cada caso de uso podem ser analisadas no diagrama de casos de uso (Figura 15, Tabela 1 e Tabela 2).

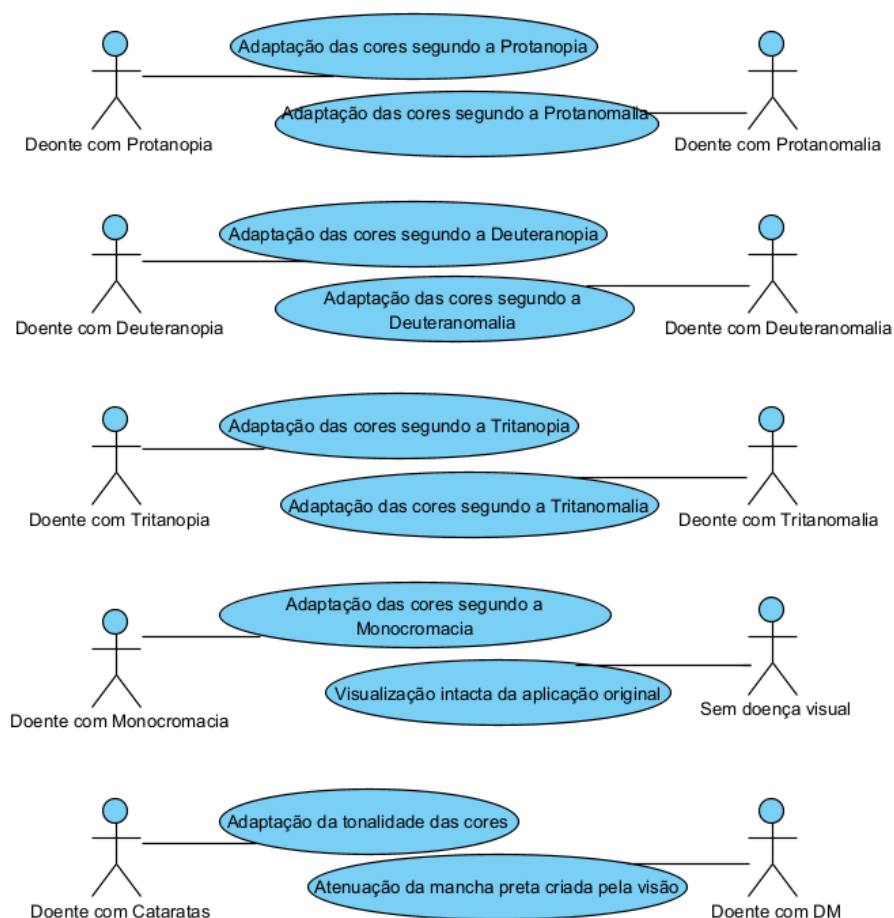


Figura 15 - Diagrama de casos de uso

Tabela 1 – Identificação dos casos de uso

ID	Nome
UC01	Adaptar as cores de uma aplicação segundo a Monocromacia
UC02	Adaptar as cores de uma aplicação segundo a Protanopia
UC03	Adaptar as cores de uma aplicação segundo a Deuteranopia
UC04	Adaptar as cores de uma aplicação segundo a Tritanopia
UC05	Adaptar as cores de uma aplicação segundo a Protanomalia
UC06	Adaptar as cores de uma aplicação segundo a Deuteranomalia
UC07	Adaptar as cores de uma aplicação segundo a Tritanomalia
UC08	Adaptar as intensidades das cores
UC09	Atenuar a mancha preta central nas imagens
UC10	Visualizar sem alterações a aplicação original

Tabela 2 - Descrição dos casos de uso

ID	Descrição
UC01	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Monocromacia
UC02	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Protanopia
UC03	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Deuteranopia
UC04	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Tritanopia
UC05	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Protanomalia
UC06	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Deuteranomalia
UC07	A aplicação deve corrigir as cores presentes na interface gráfica segundo os filtros necessários para a Tritanomalia
UC08	A aplicação deve conseguir regular as intensidades das cores para que aumente a facilidade de compreensão de um doente com Cataratas
UC09	A aplicação deve conseguir atenuar as manchas pretas vistas pelos doentes com DM
UC10	A aplicação não deve fazer qualquer tipo de alteração a nível de cor, ou suas tonalidades, nem alterar imagens.

Para além dos atores e respetivos casos de uso explicados, considera-se também ator do projeto qualquer desenvolvedor que tenha interesse ou necessidade em adaptar o seu trabalho segundo as deficiências visuais presentes nesta biblioteca, e que por essa razão recorra a ela para conseguir melhorar a acessibilidade da sua aplicação.

4.3 Requisitos Não Funcionais

São considerados os requisitos não funcionais aqueles que estão relacionados com o bom funcionamento do projeto e a sua qualidade. Estes podem não estar diretamente ligados às funcionalidades descritas anteriormente.

Assim, e seguindo o modelo FURPS+, os requisitos não funcionais relativos ao projeto são descritos segundo as categorias:

- Funcionalidade – relativa à funcionalidade da biblioteca em que devem ser registados todos os erros e posteriormente persistidos no sistema para que possam ser analisados e corrigidos;
- Usabilidade – relativa à interação entre o utilizador e o software é expectável que a biblioteca venha aumentar a facilidade de acesso dos utilizadores, que padeçam de

uma das três doenças agregadas, nas aplicações que utilizem a biblioteca implementada. Assim, para que se consiga medir este requisito, a biblioteca será inserida em aplicações já implementadas para que, com recurso a pessoas com deficiências visuais a quem se destina, se realizem sessões de teste;

- Confiabilidade – relativa à capacidade de o software atuar sem qualquer falha em diversas condições, à sua integridade, e em conformidade com o que pretendido. É expectável que a biblioteca implementada consiga adaptar a interface gráfica segundo as três doenças propostas e que para cada doença o faça de forma exímia;
- Suportabilidade – inclui categorias como: adaptabilidade, em que se espera que o software desenvolvido seja adaptável a mudanças de negócio, como por exemplo o aumento de novos requisitos; configurabilidade, em que seja possível o utilizador personalizar o software consoante as suas necessidades, como por exemplo conseguir inserir o tipo de daltonismo que o caracteriza para que desta forma sejam utilizados os filtros que melhor se adequem; manutenibilidade, de forma a que se mantenha a melhoria contínua do software, realizando correções de erros, alterações de requisitos, aumentando o nível de qualidade e consecutiva prevenção de futuros erros;
- Desempenho – relativo à medição do tempo de respostas ou tempo de recuperação, espere-se que não exista uma diferença significativa entre o desempenho da aplicação que utiliza a biblioteca desenvolvida e a mesma aplicação sem o recurso da biblioteca.

4.4 Design

O objetivo desta seção é planejar e desenhar a solução do projeto. Para isso, inicialmente são listadas as classes que constituem a biblioteca e as relações entre elas. Para melhor visualização global, foi realizado um diagrama de classes (Figura 16), onde se podem verificar algumas das classes alteradas. O diagrama completo pode ser visto no Anexo A do documento.

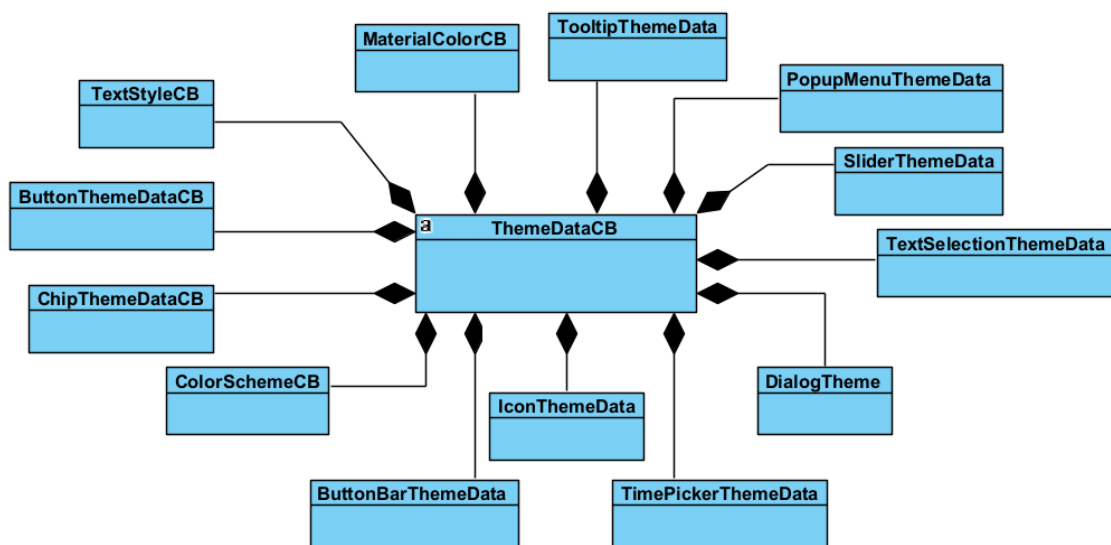


Figura 16 - Diagrama de Classes

A classe ThemeDataCB é a classe principal da biblioteca que agrega toda a lista de classes pertencentes aos diferentes componentes visuais. É uma classe filha que estende a classe ThemeData, que por sua vez define a configuração do tema visual para uma MaterialApp [41]. Esta última representa uma aplicação Flutter que segue as diretrizes de design Material da Google. Ou seja, esta classe é utilizada para configurar toda a aparência de uma aplicação em Flutter [42].

A classe principal ThemeDataCB agrega todas as classes, pertencentes à classe ThemeData, que foram necessárias alterar para que as cores nelas presentes sejam modificadas segundo os diferentes tipos de daltonismo. Assim as classes alteradas podem ser divididas em dois grupos:

1. As classes em que foi necessário a criação de uma nova classe, que entendia a classe base;
2. As classes em que se conseguiu utilizar o método copyWith para alterar as cores nela presentes. O método copyWith permite recriar uma instância de um objeto com novos atributos, mantendo os valores antigos dos mesmo caso não se alterem.

Na Figura 16 apenas são representados alguns exemplos das classes mencionadas para que o diagrama fosse de melhor leitura, a versão completa encontra-se no Anexo A. Assim, as classes criadas através da extensão das classes base são:

- *AppBarThemeCB*: Substitui os valores padrão das propriedades visuais do *Widget AppBar* [43];
- *ButtonThemeDataCB*: Substitui os valores padrão das propriedades visuais do *Widget ButtonTheme* [44];
- *ChipThemeDataCB*: Substitui os valores padrão das propriedades visuais do *Widget ChipTheme* [45];

- *ColorSchemeCB*: Conjunto de cores com base na especificação do Material, usado para a configuração de componentes [46];
- *InputDecorationThemeCB*: Substitui os valores padrão das propriedades visuais dos *Widgets InputDecoration*, *TextField* e *TextFormField* [47];
- *MaterialColorCB*: Define uma única cor através de um conjunto de dez cores [48];
- *TabBarThemeCB*: Substitui os valores padrão das propriedades visuais do *Widget TabBar* [49];
- *TextStyleCB*: Substitui os valores padrão das propriedades visuais do *Widget TextStyle* [50];
- *TextThemeCB*: Substitui os valores padrão das propriedades visuais do *Widget TextTheme* [51];
- *ToggleButtonThemeDataCB*: Substitui os valores padrão das propriedades visuais do *Widget ToggleButtons* [52].

Por outro lado, as classes onde foi possível a utilização do método *Copy With* são:

- *IconThemeData*: Substitui os valores padrão das propriedades visuais do *Widget Icon* [53];
- *MaterialBannerThemeData*: Substitui os valores padrão das propriedades visuais do *Widget MaterialBanner* [54];
- *BottomAppBarTheme*: Substitui os valores padrão das propriedades visuais do *Widget BottomAppBar* [55];
- *BottomNavigationBarThemeData*: Substitui os valores padrão das propriedades visuais do *Widget BottomNavigationBar* [56];
- *BottomSheetThemeData*: Substitui os valores padrão das propriedades visuais do *Widget BottomSheet* [57];
- *ButtonBarThemeData*: Substitui os valores padrão das propriedades visuais do *Widget ButtonBar* [58];
- *CardTheme*: Substitui os valores padrão das propriedades visuais do *Widget Card* [59];
- *NoDefaultCupertinoThemeData*: Especificação de estilos sem valores padrão para propriedades não especificadas [60];
- *DataTableThemeData*: Substitui os valores padrão das propriedades visuais do *Widget DataTable* [61];
- *DialogTheme*: Substitui os valores padrão das propriedades visuais do *Widget Dialog* [62];
- *DividerThemeData*: Substitui os valores padrão das propriedades visuais do *Widget Divider* [63];
- *FloatingActionButtonThemeData*: Substitui os valores padrão das propriedades visuais do *Widget FloatingActionButton* [64];
- *NavigationRailThemeData*: Substitui os valores padrão das propriedades visuais do *Widget NavigationRail* [65];
- *PopupMenuThemeData*: Substitui os valores padrão das propriedades visuais dos *Widgets PopupMenuItem* e *PopupMenuDivider* [66];

- *SliderThemeData*: Substitui os valores padrão das propriedades visuais do *Widget SliderTheme* [67];
- *SnackBarThemeData*: Substitui os valores padrão das propriedades visuais do *Widget SnackBar* [68];
- *TextSelectionThemeData*: Substitui os valores padrão das propriedades visuais do *Widget TextSelection* [69] *TextSelectionThemeData*;
- *TimePickerThemeData*: Substitui os valores padrão das propriedades visuais do *Widget TimePicker* [70];
- *TooltipThemeData*: Substitui os valores padrão das propriedades visuais do *Widget Tooltip* [71];
- *Typography*: Substitui os valores padrão das propriedades visuais do *Widget TextThemes* [72].

4.5 Resumo

Este capítulo serviu assim para a enumeração e descrição dos requisitos funcionais e não funcionais do projeto. Sendo que para os requisitos funcionais forma identificados os principais atores e descritos os casos de uso respetivos e para os requisitos não funcionais foi utilizado o modelo FURPS+ para a sua identificação e explicação. Por outro lado, foi também apresentado o design do projeto, e explicado a arquitetura do mesmo sendo estes uma maisvalia para a sua melhor compreensão.

5 Implementação

Após enumerar os requisitos funcionais e não funcionais e apresentar a arquitetura do projeto, este capítulo destina-se à descrição da implementação da biblioteca proposta.

5.1 Decisões tecnológicas

No que diz respeito à linguagem de programação ou *framework* para a qual seria desenvolvida a biblioteca foram consideradas:

- React [73];
- Flutter [74];
- Electron [75];
- React Native [76].

Esta escolha teve por base a seleção de *framework* para desenvolvimento *web*, *desktop* e *mobile*. Segundo a plataforma Statista, são considerados como as duas melhores *frameworks* para desenvolvimento *mobile* o Flutter e o React Native [77] e a mais utilizada para desenvolvimento *web* a React.js [78]. Por outro lado, para o desenvolvimento *desktop* foi considerada a Electron uma vez que é uma *framework* utilizada em diversas aplicações conhecidas como por exemplo VSCode, Facebook Messenger e Slack [79], o que demonstra algum apreço entre os desenvolvedores.

Outro aspeto que foi tido em conta nas *frameworks* a serem consideradas, foi a respetiva popularidade na plataforma GitHub. A *framework* com menor popularidade, mas mesmo assim com um total de 98 200² estrelas, é a Electron. Já a *framework* com o maior número de estrelas e por isso mais popular entre os desenvolvedores é a React com o total de 176 000³.

² Valor retirado da plataforma GitHub no dia 05/10/2021.

³ Valor retirado da plataforma GitHub no dia 05/10/2021.

Na Tabela 3 estão representados os critérios analisados para a escolha da *framework* a ser utilizada no desenvolvimento do projeto.

Tabela 3 - Critérios de seleção da *framework*

	React	Flutter	Electron	React Native
Linguagem de desenvolvimento	JavaScript	Dart	C++ e JavaScript	JavaScript
Usabilidade	Nativa	Nativa	Nativa	Nativa
Código aberto	Sim	Sim	Sim	Sim
Comunidade	A maior	Grande	Grande	Grande
Plataformas de Suporte	Web	Android, iOS, Fuchsia OS, Web, Windows, macOS, Linux, Universal Windows Platform (UWP)	Windows, macOS, Linux	Android, iOS

Fazendo a análise da Tabela 3 é possível validar que, para as linguagens de desenvolvimento, tanto a React como a React Native usam JavaScript, Flutter usa Dart e Electron usa C++ e JavaScript. Embora a Flutter seja a única plataforma a não utilizar JavaScript, sendo esta mais conhecida e popular, Dart acaba por ter semelhanças a Java ou C# por partilhar conceitos da programação orientada a objetos e como tal é considerada de fácil aprendizagem e adaptação.

Por outro lado, no que diz respeito à usabilidade, todas as aplicações desenvolvidas em qualquer uma das quatro *frameworks* apresentam características nativas, e são também todas elas de código aberto, não sendo assim necessário qualquer custo para desenvolver aplicações.

A nível do critério relativo à comunidade, a que tem um maior destaque é a React, sendo a de menor destaque, mas mesmo assim com uma comunidade grande, a Electron. É de destacar o crescimento neste critério da *framework* Flutter que em apenas dois anos ultrapassou a React Native e a Electron. Este critério foi avaliado segundo o nível de popularidade na plataforma GitHub como já referenciado anteriormente.

Relativamente ao critério das plataformas de suporte existe uma clara distinção do Flutter, uma vez que mostra um suporte mais amplo, ainda que para desktop (Windows, Linux e macOS) esteja numa fase beta e para UWP numa fase alfa.

Assim, com base na informação apresentada, por ser uma tecnologia recente e por isso suscitar interesse na sua aprendizagem, a Flutter foi a *framework* selecionada para o desenvolvimento do projeto.

5.2 Flutter

Flutter é uma ferramenta de desenvolvimento de interface e de código aberto desenvolvida pela Google e lançada em 2017. Tem por base a linguagem Dart e atualmente suporta Android, iOS, Windows, MacOS, Linux, Fuchsia OS e *Web* [80]. Apesar de ser uma ferramenta recente, o Flutter tem vindo a ter uma adoção exponencial por várias pequenas e médias empresas bem como multinacionais. Entre algumas das aplicações Flutter já desenvolvidas destacam-se, por exemplo: Xianyu do grande comércio eletrónico Alibaba, My BMW app, NU do banco Nubank, várias aplicações da empresa Tencent (AITeacher, Now Live e K12), Google Assistant e Square [81].

A biblioteca de *Widgets* é a principal forma de desenvolvimento em Flutter e como tal é necessário perceber o que é um *Widget*, como é que ele se comporta com os restantes componentes e como gerir os estados da aplicação com *Stateful Widgets*.

5.2.1 Widget

Um *Widget* em Flutter é um componente ou um conjunto de componentes de uma interface, ou seja, tudo numa UI é um *Widget*. Descrevem a sua configuração e estado num determinado momento e podem ser considerados *Widgets* imutáveis, sem estado (*Stateless Widget*) [82] ou mutáveis, com estado (*Stateful Widget*) [83]. Os *Stateful Widgets* são compostos por um objeto de estado que pode variar ao longo do tempo e que com isso irá atualizar a sua descrição gráfica.

Embora os *Widgets* mutáveis e imutáveis sejam os mais comuns e mais utilizados, existem também os *InheritedWidgets* [84], cujo objetivo é guardar informação e propagá-la de maneira eficiente numa árvore de *Widgets*. Os filhos *Widgets* de um *InheritedWidget* conseguem aceder à sua informação através do *BuildContext*.

Sendo os *Stateless Widget* e os *Stateful Widgets* os mais comuns, é importante perceber os seus ciclos de vida. Assim, é apresentado na Figura 17 o diagrama de atividades de um *Stateless Widget* e na Figura 18 o diagrama de atividades de um *Stateful Widget*.

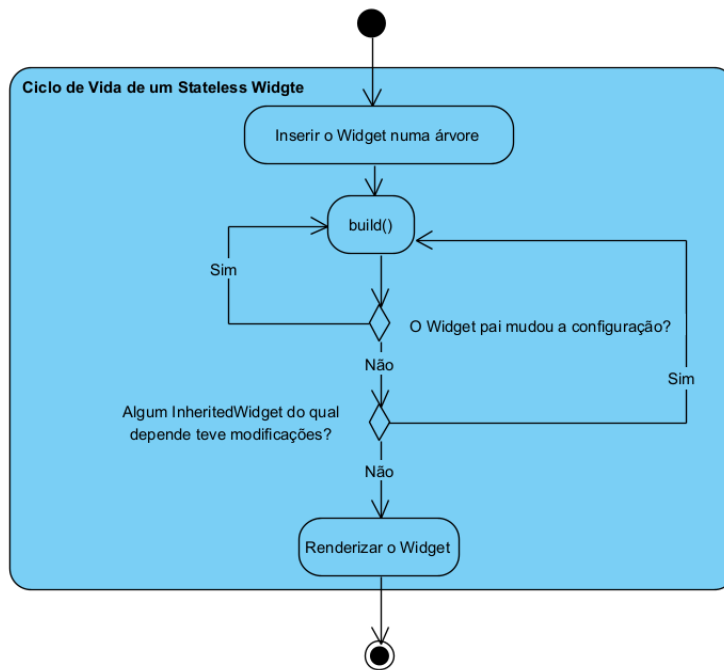


Figura 17 - Diagrama de atividades Stateless Widget

Um Stateless Widget começa por ser inserido numa árvore de Widgets e de seguida o Flutter chama o build desse Widget. De seguida é validado se o pai do Widget sem estado teve alguma alteração na sua configuração e caso esta resposta seja positiva o Flutter faz novamente build ao Stateless Widget; caso seja negativa, valida se existiu alguma alteração no InheritedWidget do qual depende (é de notar que o StatelessWidget pode ou não ter algum Widget pai que seja InheritedWidget). Se tiverem existido alterações é feito novamente o build do Stateless Widget; caso não tenham existido alterações é feita a renderização do Widget final [82].

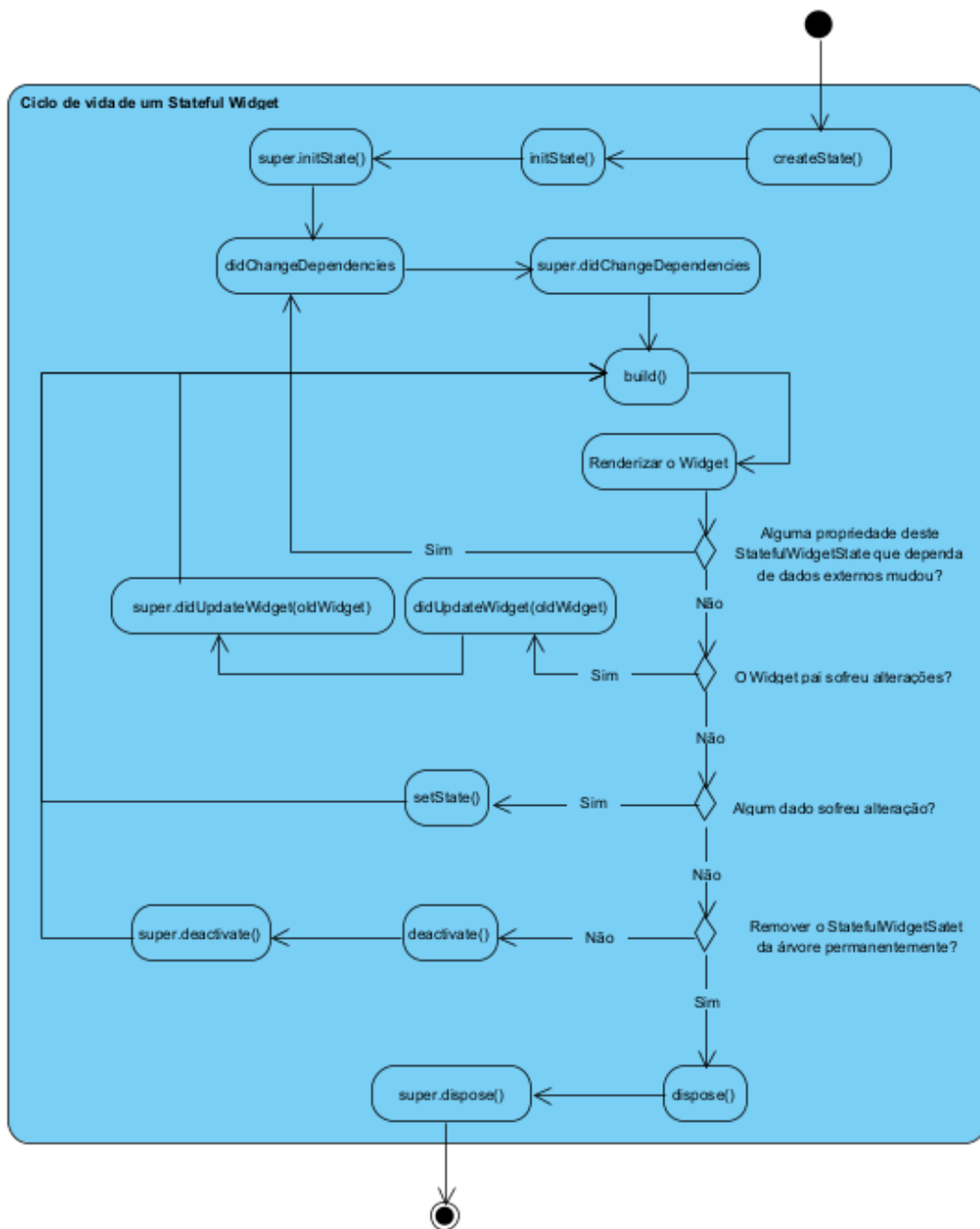


Figura 18 - Diagrama de atividades Stateful Widget

Para o ciclo de vida de um Stateful Widget é necessário em primeiro lugar criar um objeto *State* e, a partir daí, todas as ações que se seguem estão no objeto *StatefulWidgetState* e não no Widget propriamente dito. Esse estado representa informações que podem ser lidas de forma síncrona quando o Widget é construído e que podem ser alteradas ao longo do seu tempo de vida. Após a criação do estado é necessário inicializá-lo. Caso o desenvolvedor pretenda iniciar dados ou realizar subscrições a *streams* ou a notificações, é necessária a criação de um *override* do método *initState()* e realizar a chamada do *super.initState()*. Caso não exista qualquer tipo de alterações, o Flutter chama o *initState()*. A iniciação do estado é apenas realizada uma única

vez, seguindo-se a chamada do método *didChangeDependencies()*, que, como indica o nome, valida se existem alterações na dependência do estado. Após a validação é feito o *build()* do *Stateful Widget* e é realizada a sua renderização. Caso tenham existido mudanças nas propriedades do *StatefulWidgetState* que dependam de dados externos é chamado novamente o método *didChangeDependencies()*. Por outro lado, caso tenham existido alterações no *Widget* pai, é chamado o método *didUpdateWidget()* para que a reconstrução do *StatefulWidget* seja feita. É também validado se algum dado do *StatefulWidgetState* sofreu alterações e caso a resposta seja positiva é chamado o método *setState()* para atualizar o *Widget*. Por fim, é validado se o *StatefulWidget* é retirado da árvore de *Widgets* e caso a resposta seja negativa então é realizada a chamada do método *deactivate()* que muda o *StatefulWidgetState* para outro lugar na árvore. Caso a resposta seja positiva, é realizada a chamada do método *dispose()*. De notar que após a chamada de cada método como o *didUpdateWidget()*, *setState()* ou *deactivate()*, método *build()* do *StatefulWidget* é sempre chamado e é realizada a sua renderização [83].

5.2.2 ThemeData

Assim, e após a compreensão dos ciclos de vida de um *Stateless* e *Stateful Widget*, passamos para a compreensão da classe principal do nosso projeto. *ThemeData* [42] é a classe principal, uma vez que, como já explicado na secção 4.4, é a classe responsável pela alteração de todo o tema de uma aplicação em Flutter. Assim, os *Widgets* que devem estar alinhados com o tema geral em termos de aparência, podem obter as configurações necessárias através do método *Theme.of*. Os *Widget Material* dependem maioritariamente das propriedades *colorScheme* e *textTheme* do *ThemeData*, que tem a garantia de não ter valor nulos.

O método *Theme.of* encontra o valor *ThemeData* especificado para o ascendente *BuildContext* [85] mais próximo. Ou seja, é possível aceder ao *ThemeData* de cada *Widget* através do seu *BuildContext* e foi esta a razão pela qual na biblioteca desenvolvida foram criadas ou alteradas todas as classes pertencentes à *ThemeData* que manipulassem cores. Desta forma conseguimos fazer a mudança segundo o tipo de daltonismo a todos os *Widgets* presentes nas aplicações que utilizem a biblioteca, manipulando apenas valores seletivos da *ThemeData*.

A classe *BuildContext* é um identificador para a localização de um *Widget* numa árvore de *Widgets*, e apresenta um conjunto de métodos que podem ser usados através do método *build* dos *Stateless Widgets* e dos *States* dos *Stateful Widgets*. Cada *Widget* tem o seu próprio *BuildContext* que se torna o pai do *Widget* retornado pela função *StatelessWidget.build* ou *State.build*. Através do *BuildContext* consegue-se obter informações como: o tamanho do ecrã do dispositivo; orientação do dispositivo; definições de acessibilidade entre outras [85].

É possível verificar no código apresentado na Listagem 1 um exemplo da explicação anterior.

```

import 'package:flutter/material.dart';

const Color darkBlue = Color.fromARGB(255, 18, 32, 47);

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      theme: ThemeData.dark().copyWith(
        scaffoldBackgroundColor: darkBlue,
        backgroundColor: Colors.lightBlue,
      ),
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        body: Center(
          child: ThemeOfContextSample(),
        ),
      ),
    );
  }
}

class ThemeOfContextSample extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Container(
      alignment: Alignment.center,
      color: Theme.of(context).backgroundColor,
      width: 200,
      height: 200,
      child: Text(
        'Hello, World!',
        style: Theme.of(context).textTheme.headline2,
        textAlign: TextAlign.center,
      ),
    );
  }
}

```

Listagem 1 - Exemplo do método Theme.of para modificar a interface

No Widget ThemeOfContextSample é possível validar que o desenvolvedor vai preencher a cor de fundo do Widget Container através da backgroundColor que se encontra disponível usando o BuildContext do Widget e o método Theme.of que devolve a instância ThemeData do mesmo contexto. Este ThemeData é criado quando da instanciação do Widget MaterialApp, que representa uma *app* que segue as linhas de orientação do design Material, e nele se encontra

a definição da `backgroundColor`. Tendo em conta que no `ThemeData` é possível definir a tipografia de uma aplicação Flutter, o estilo do texto “Hello World” também é fornecido através do método `Theme.of`, que neste caso é o `headline2`.

5.2.3 *Plugin Color_Blindness*

Ao longo do desenvolvimento do projeto foi encontrado apenas um *plugin* Flutter existente relacionado com o daltonismo, com o nome `color_blindness` e com a versão 0.1.0 lançada a 23 de fevereiro de 2021 [86].

Assim, no desenvolvimento do projeto foi utilizado este *plugin* que tem como principal objetivo a alteração do esquema de cores segundo um tipo de daltonismo. O *plugin* está preparado para os diferentes tipo de daltonismo: Protanopia, Deuteranopia, Tritanopia, Protanomalia, Deuteranomalia, Tritanomalia e Monocromacia. Ao ser utilizado simulará o tipo de daltonismo selecionado alterando todas as cores do esquema de cores a que seja chamado.

```
import
'package:random_colorscheme/color_blindness_color_scheme.dart';

const primary = const Color(0xff9f0042);
// indirect way
colorBlindness(primary, ColorBlindnessType.tritanopia);

// direct way
tritanopia(primary);
```

Listagem 2 - Exemplo de utilização do *plugin* `color_blindness`

A Listagem 2 demonstra um excerto de código a utilizar o *plugin* apresentado. Através da invocação do método `colorBlindness`, que requer uma cor para alterar e um tipo de daltonismo do qual a alteração da cor depende, consegue-se uma manipulação de cores em Flutter simples e concisa.

Assim, foi com o recurso a este método que foi delineada a restante implementação da biblioteca deste projeto, alterando todos os objetos cor presentes nas diferentes classes associadas à `ThemeData`.

5.2.4 *Aplicação da transformação de cores em Imagens*

Uma das lacunas encontrada na biblioteca `color_blindness` foi a falta de métodos para transformação de imagens e como tal a biblioteca desenvolvida neste projeto tenta colmatar essa mesma falta. Assim, foi criada também a transformação de imagens segundo os diferentes tipos de daltonismo. Começou-se por realizar a leitura de uma imagem, guardando a

informação de todos os seus pixels. De seguida, e para cada informação do pixel, foi retirada a sua cor e foi invocado o método `colorBlindness`. Após a transformação de todas as cores de todos os pixels da imagem, é feita a reconstrução da mesma já com as cores adaptadas ao tipo de daltonismo requerido. O processamento da imagem é feito através do recurso ao *FutureBuilder Widget* [87], que se constrói com base no estado mais recente de um objeto *Future* que representa uma computação que é executada de forma assíncrona. Assim, a transformação das cores da imagem é feita nesse objeto e durante o processo de transformação das cores e da reconstrução da imagem é mostrado um indicador de carregamento ou outro *Widget* que o desenvolvedor pretenda apresentar aos utilizadores da sua aplicação. Após ser feita a transformação das cores, a *Future* é terminada e o *FutureBuilder* é reconstruído mostrando o resultado final que acaba por ser a imagem já com o filtro do respetivo tipo de daltonismo. É possível observar um exemplo desta explicação na Figura 19.



Figura 19 - Exemplo de transformação de imagens

Na Figura 19 é possível ver a imagem original de um conjunto de lápis de cor (destacada acima das restantes), bem como as transformações da mesma segundo o algoritmo desenvolvido e explicado anteriormente. Assim, na parte inferior da imagem e numa ordenação da esquerda para a direita temos:

1. A transformação da imagem original para pessoas que padeçam de Monocromacia, pessoas que apenas distinguem as tonalidades entre o preto e o branco por terem anomalias ou ausência dos três cones;
2. A transformação da imagem original para pessoas que padeçam de Deuteranopia, ausência na retina de cones verdes;
3. A transformação da imagem original para pessoas que padeçam de Protanopia, ausência na retina de cones vermelhos;
4. A transformação da imagem original para pessoas que padeçam de Tritanopia, ausência de cones azuis.

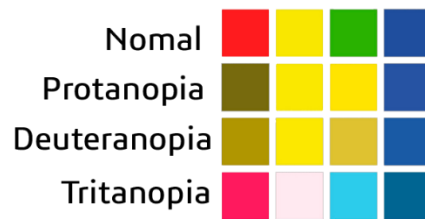


Figura 20 - Comparação de cores nos diferentes tipos de daltonismo

A Figura 20 apresenta a comparação entre os três tipos de daltonismo do exemplo anterior perante diferentes cores de origem. Conclui-se que existe uma semelhança grande entre as tonalidade e transformação de cores entre a Figura 19 e a Figura 20.

5.3 Resumo

Este capítulo serviu assim para a identificação das tecnologias possíveis, realizando comparação entre elas para que com um conjunto de critérios estabelecidos tenha sido escolhida a *framework* Flutter para o desenvolvimento do projeto. Foi também realizada a explicação pormenorizada da tecnologia escolhida, dos componentes alterados, das classes criadas e do mecanismo criado para a transformação de imagens. Por fim foi ainda dado um exemplo da transformação de imagens e a sua posterior renderização.

6 Experiências e Avaliação

Este capítulo tem como objetivo descrever as experiências e avaliações realizadas ao projeto desenvolvido, e como tal são descritas as abordagens utilizadas para a avaliação da biblioteca, bem como as experiências realizadas e os resultados obtidos.

6.1 Abordagem da avaliação

A abordagem seguida no sentido de avaliar a biblioteca desenvolvida pode ser separada em duas vertentes diferentes segundo os dois tipos de atores definidos na secção 4.2.

Por um lado, temos a avaliação e testagem da biblioteca segundo a perspectiva dos desenvolvedores, que têm interesse em que a utilização da mesma seja de fácil compreensão, de simples utilização, abrangente a todos os tipos de daltonismo e de rápida integração com as aplicações.

Por outro, temos a avaliação e testagem por parte de pessoas com diferentes tipos de daltonismo, em que a biblioteca é inserida numa aplicação de testes preparada com vários componentes, imagens e cenários comuns onde o daltónico apresente dificuldades.

6.2 Experiências Realizadas

Segundo o lado dos desenvolvedores, a avaliação foi realizada de uma forma informal ao longo da semana de 19 a 25 de setembro de 2021 por parte de colegas de curso, uma vez que o tempo foi escasso nesta fase do projeto.

Estes integraram a biblioteca em aplicações Flutter para avaliarem a sua compreensão, utilização e integração. Foram também utilizadas as aplicações Flutter destes participantes para testar todos os tipos de daltonismo e assim, realizada a validação da transformação de cor em todos os componentes e para os diferentes tipos de daltonismo.

Segundo o lado das pessoas que padecem de daltonismo as suas avaliações também foram realizadas de uma forma informal pela mesma razão da avaliação do lado dos desenvolvedores.

Assim foi criada uma aplicação base em Flutter que continha diferentes componentes de diferentes cores, nomeadamente: textos, botões, *dropdowns*, *sliders*, ícones e imagens, para que desta forma fossem testados diferentes componentes alterados. A Figura 21 representa parte da aplicação de testes original, sendo que as adaptações para os diferentes tipos de daltonismo encontram-se no Anexo B.

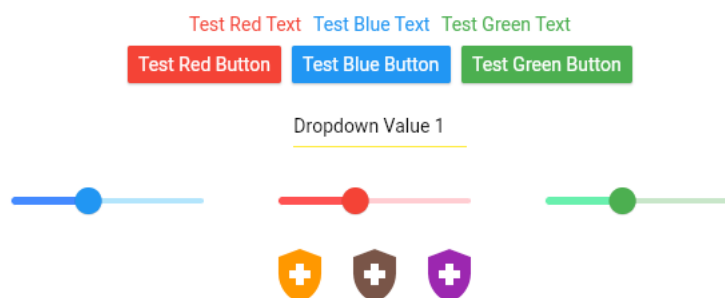


Figura 21 - Aplicação de teste original

Esta avaliação foi realizada por dois participantes que padecem de daltonismo, sendo que o participante A padece de Protanomalia e o participante B de Tritanomalia, ao longo da semana entre 10 e 16 de outubro de 2021.

Reconhecendo que a natureza destes testes é apenas qualitativa, e apenas para um número reduzido de participantes, seria uma mais-valia ao projeto desenvolvimento de testes mais pormenorizados. Como tal, seria necessário a criação de formulários, com as questões indicadas e enquadradas nos parâmetros a serem avaliados para os dois grupos distintos. Assim, para um conjunto mais alargado de participantes e com as conclusões e avaliações perante os questionários respondidos, existiria uma avaliação quantitativa da biblioteca.

6.3 Resultados Obtidos

Os desenvolvedores que avaliaram o projeto desenvolvido nas suas aplicações de testes consideraram a biblioteca de fácil compreensão e utilização e não tiveram qualquer impedimento ou falha na sua integração com a aplicação de teste.

Para a avaliação feita em relação à utilização das aplicações para todos os tipos de daltonismo, estas tiveram os resultados esperados, não existindo qualquer tipo de erro na diferenciação dos tipos de daltonismo e a transformação de cores foi feita segundo as cores esperadas perante o estudo feito anteriormente.

Segundo a avaliação realizada pelo participante A, os nomes das classes criadas a partir das originais do ThemeData deveriam em vez da abreviatura “CB” no final do nome da classe ter escrito por extensão *ColorBlind* para que fosse mais rápida e fácil a interpretação da mesma.

Segundo a avaliação realizada pelo participante B, o facto de não existir documentação e testes unitários para a biblioteca pode ser um fator negativo. Isto porque não é dada ao desenvolvedor qualquer garantia de qualidade e que a leitura do código, apesar de simples, pode não ser fácil a todos os desenvolvedores.

Para ambos os participantes, como as classes criadas seguem o estilo das mesmas classes do Flutter, tonar-se muito fácil perceber o que é que cada uma representa, para que serve e que parâmetros necessita.

Para a avaliação feita pelas pessoas que padecem de daltonismo foi feita a comparação das imagens e componentes originais com os transformados segundo o filtro de daltonismo que caracteriza cada participante e a conclusão nos dois casos foi bastante positiva. Ou seja, a visão perante a aplicação de testes original e a aplicação com a biblioteca perante o tipo de daltonismo de cada uma era muito semelhante. De notar que para o tipo da tricomacia, as irregularidades nos cones podem variar e como tal podem afetar mais ou menos a visão perante a cor do cone afetado. Por exemplo, para o participante A, apesar de na grande parte dos componentes e imagens as cores se assemelharem todas quando comparadas com e sem filtro para a Protanomalia, para as tonalidades entre o amarelo e o verde existiam algumas diferenças ainda que muito reduzidas e vistas apenas em esforço. O mesmo aconteceu na avaliação por parte do participante B, neste caso para a Tritanomalia, e para as tonalidades azuis.

6.4 Resumo

Neste capítulo foram descritas as abordagens de avaliação para a biblioteca desenvolvida que podem ser divididas em duas visões segundos os atores identificados. Por um lado, existiu a experimentação e avaliação dos desenvolvedores que utilizaram a biblioteca desenvolvida nas suas aplicações de testes, avaliando assim a sua dificuldade, utilização e integração em aplicações Flutter. Por outro, existiu a experimentação e avaliação por parte de pessoas daltónicas, também através de uma aplicação de testes que apresentava diferentes componentes e um conjunto de imagens com as alterações de cores necessárias segundo o seu tipo de daltonismo.

7 Conclusões

Neste capítulo são apresentadas as conclusões do projeto desenvolvido, nomeadamente, os objetivos alcançados, as melhorias que podem ter sido em consideração no futuro bem como novos objetivos a serem adicionados e ainda a apreciação final do projeto desenvolvido.

7.1 Objetivos Alcançados

O objetivo principal deste projeto visava o desenvolvimento de uma biblioteca direcionada para a criação de interface gráfica com o utilizador para pessoas com algum tipo de distúrbio visual, garantindo ser uma ajuda aquando da interação com aplicações por parte destas pessoas.

Segundos objetivos definidos na secção 1.3 a Tabela 4 demonstra o estado de cada um.

Tabela 4 - Estado final dos objetivos do projeto

Objetivo	Estado
Estudar diferentes distúrbios visuais	Acabado
Identificar as técnicas e tecnologias já existentes no mercado	Acabado
Desenhar a solução a ser desenvolvida	Acabado para uma doença
Desenvolver a solução segundo as boas práticas da Engenharia	Acabado para uma doença
Avaliar a solução final	Parcialmente acabado para uma doença

Relativamente ao primeiro ponto e aquele que se destinava ao estudo dos diferentes distúrbios visuais foram identificados dois tipos de deficiências visuais: oculares (DVO), as que surgem diretamente no olho e corticais (DVC), as que podem surgir na estrutura neural ou região cerebral.

Para o primeiro grupo foram identificados os erros refrativos (Miopia, Hipermetropia e Astigmatismo), as Cataratas, a Discromatopsia, a Retinopatia Diabética e a Degenerescência Macular.

Para o segundo grupo foram identificadas doenças como a Glaucoma, Diplopia e Papiledema.

Depois de identificadas as doenças mais comuns que afetam a visão do ser humano foram selecionadas três para o desenvolvimento do projeto, sendo selecionadas as Cataratas, Discromatopsia e Degenerescência Macular.

No que diz respeito ao segundo objetivo delineado, foram identificados alguns algoritmos que auxiliam os designers na seleção das paletas de cores para os seus projetos; aplicações móveis

que auxiliam a identificação do tipo de daltonismo e ainda um conjunto de bibliotecas e *plugins* que simulam diferentes tipos de daltonismo.

Sobre os últimos três objetivos, relativos ao desenho da solução, desenvolvimento e avaliação apenas foi possível a concretização dos objetivos para a doença da discromatopsia e espera-se aumentar o número de doenças da biblioteca em trabalho futuro.

7.2 Trabalho Futuro

Ao nível do trabalho futuro, espera-se o aumento da biblioteca segundo a doença das Cataratas e Degenerescência Macular no sentido de tornar a biblioteca mais abrangente em termos de deficiências visuais suportadas. É de contemplar também a possibilidade de serem estudadas outras doenças visuais e serem também estas inseridas na biblioteca.

Seria também interessante adicionar a biblioteca desenvolvida à biblioteca encontrada e utilizada no trabalho desenvolvido com o principal objetivo de se complementarem e tornar assim a biblioteca *color_blindness* mais rica e completa, contribuindo assim para a possibilidade de existir uma biblioteca padrão na comunidade Flutter para desenvolver aplicações considerando deficiências visuais.

7.3 Apreciação final

O desenvolvimento deste projeto foi uma mais-valia tanto a nível profissional, como a nível pessoal e espero que venha a ajudar também muitos desenvolvedores que tenham o cuidado de adaptar as suas aplicações para pessoas com distúrbios visuais e que assim utilizem a biblioteca desenvolvida.

A nível profissional, este projeto permitiu-me aplicar os conhecimentos adquiridos ao longo da licenciatura e mestrado em Engenharia Informática, permitindo aprimorar as capacidades de desenvolvimento de software, aprender uma *framework* nova, Flutter, e ainda melhorar a gestão de projetos e planeamento do mesmo.

Ao longo do desenvolvimento, a gestão dos contratempos, a receção das críticas e melhorias, todas as decisões tomadas e o estudo sobre deficiências visuais foram também uma mais-valia para o meu crescimento pessoal.

Por fim considero o resultado do projeto satisfatório tendo em conta os recursos disponíveis.

Referências

- [1] INE, “No Title,” 2019. https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaques&DESTAQUESdest_boui=354447153&DESTAQUEStema=55483&DESTAQUESmodo=2 (accessed Nov. 29, 2020).
- [2] Samsung, “Samsung,” 2021. <https://www.samsung.com/pt/support/tv-audio-video/smart-tv-js9000-como-ativar-o-guia-de-voz-para-utilizadores-com-dificuldades-visuais/>.
- [3] “NOS.” <https://www.nos.pt/particulares/ajuda/equipamentos-servicos/televisao/manuais-e-equipamento/Documents/ManualComandoPorVoz.pdf>.
- [4] ACAPO, “ACAPO,” 2021. <http://www.acapo.pt/deficiencia-visual/perguntas-e-respostas/deficiencia-visual>.
- [5] S. Nicola, “Análise de Valor.”
- [6] “NCBI.” <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1479975/>.
- [7] INE, “Análise de População com Deficiência - Censos 2001,” *Inst. Nac. Estatística*, p. 10, 2002.
- [8] U. Federal, “UNIVERSIDADE FEDERAL DE SÃO CARLOS PROGRAMA DE PÓS-GRADUAÇÃO EM EDUCAÇÃO ESPECIAL CONSULTORIA COLABORATIVA ESCOLAR NA ÁREA DA Tese apresentada ao Programa de Pós Graduação,” 2013.
- [9] J. N. Murta *et al.*, “Rede Nacional de Especialidade e de Referência em Oftalmologia,” p. 58, 2016.
- [10] G. M. MacHado, M. M. Oliveira, and L. A. F. Fernandes, “A physiologically-based model for simulation of color vision deficiency,” *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 6, pp. 1291–1298, 2009, doi: 10.1109/TVCG.2009.113.
- [11] “Sociedade Portuguesa Oftalmologia.” https://spoftalmologia.pt/perguntas_frequentes/problemas-e-doencas-oculares/.
- [12] “Psicologia , Saúde e Doenças ISSN : 1645-0086 Sociedade Portuguesa de Psicologia da Saúde Portugal Silva , Isabel ; Pais-Ribeiro , José Aspectos psicossociais e qualidade de vida da degenerescência macular relacionada com a idade Como citar este artigo Nú,” 2006.
- [13] C. André, N. Em, and Y. Fi, “Projeto Diretrizes Tratamento Hipertireoidismo da Fase Aguda do Acidente Vascular Cerebral Projeto Diretrizes,” pp. 1–10, 2002.
- [14] Maiti and Bidinger, “DALTONISMO E AS DIFERENTES PERCEPÇÕES DE CORES,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 1981.

- [15] I. Madeira, “Degenerescência Macular Relacionada com a Idade: a propósito de um caso clínico,” p. 76, 2019.
- [16] W3C, “Web Content Accessibility Guidelines,” 2021. <https://www.w3.org/WAI/standards-guidelines/wcag/>.
- [17] R. J. De Araújo, J. Cesar, D. Reis, and R. Bonacin, “Understanding interface recoloring aspects by colorblind people : a user study,” *Univers. Access Inf. Soc.*, vol. 0, no. 0, p. 0, 2018, doi: 10.1007/s10209-018-0631-7.
- [18] L. Troiano, “Adapting palettes to color vision deficiencies by genetic algorithm,” no. January 2008, 2014, doi: 10.1145/1389095.1389291.
- [19] S. E. E. Profile and S. E. E. Profile, “Web Page Textual Color Contrast Compensation for CVD Users Using Optimization Methods Web page textual color contrast compensation for CVD users using optimization methods,” no. December, 2013, doi: 10.1007/s10852-013-9239-3.
- [20] L. Jefferson, R. Harvey, H. U. Machine, and S. Human, “Accommodating Color Blind Computer Users *,” pp. 40–47.
- [21] J. Huang *et al.*, “Enhancing Color Representation for the Color Vision Impaired To cite this version : HAL Id : inria-00321936 Enhancing Color Representation for the Color Vision Impaired,” 2008.
- [22] G. R. Kuhn, M. M. Oliveira, and L. A. F. Fernandes, “An Efficient Naturalness-Preserving Image-Recoloring Method for Dichromats,” vol. 14, no. 6, 2008.
- [23] Samsung, “Samsung Launches SeeColors App,” 2021. <https://news.samsung.com/global/samsung-launches-seecolors-app-for-qlcd-tv-to-support-people-with-color-vision-deficiency-cvd>.
- [24] Microsoft, “Color Binoculars,” 2021. <https://www.microsoft.com/en-us/garage/profiles/color-binoculars/>.
- [25] “Color Blind Pal,” 2021. <https://colorblindpal.com/>.
- [26] “Definições Microsoft,” 2021. <https://support.microsoft.com/pt-pt/office/alterar-a-luminosidade-o-contraste-ou-a-nitidez-de-uma-imagem-48f8f54b-3db7-4652-8928-9ace995240c7>.
- [27] “Contraste de cor Google,” 2021. <https://support.google.com/accessibility/android/answer/7158390?hl=pt>.
- [28] Apple, “Utilizar as preferências,” 2021. <https://support.apple.com/pt-pt/HT207025>.
- [29] Google, “Acessibilidade do Android,” 2021. <https://support.google.com/accessibility/android/answer/6151850?hl=pt><https://support.google.com/accessibility/android/answer/6151850?hl=pt>.
- [30] Microsoft, “Daltonismo Windows 10.” <https://support.microsoft.com/pt-br/topic/usar-filtros-de-cor-no-windows-10-43893e44-b8b3-2e27-1a29-b0c15ef0e5ce>.
- [31] Npm, “npm Color Blind,” 2021. <https://www.npmjs.com/search?q=color+blind>.

- [32] R, "colorBlindness Guide," 2021. <https://cran.r-project.org/web/packages/colorBlindness/vignettes/colorBlindness.html>.
- [33] "RGBblind," 2021. <https://github.com/interaktivarum/rgbblind/blob/master/README.md>.
- [34] "ColorBlindnessFlutter." <https://github.com/bernaferri/ColorBlindnessFlutter>.
- [35] "Colorblind - Dalton for Google Chrome," 2021. <https://chrome.google.com/webstore/detail/colorblind-dalton-for-goo/afcafnelafcgjinkaeohkalmfececool>.
- [36] "Let's get color blind," 2021, [Online]. Available: <https://chrome.google.com/webstore/detail/lets-get-color-blind/bkdgdianpkfahpkmphehigalpighjck>.
- [37] P. A. Koen *et al.*, "Fuzzy Front End : and Techniques," *Ind. Res.*, vol. pp, pp. 5–35, 1996, [Online]. Available: http://www.stevens.edu/cce/NEW/PDFs/FuzzyFrontEnd_Old.pdfNEW/PDFs/FuzzyFrontEnd_Old.pdf.
- [38] T. Woodall, "Conceptualising ' Value for the Customer ' : An Attributional , Structural and Conceptualising ' Value for the Customer ' : An Attributional , Structural and Dispositional Analysis," no. January 2003, 2015.
- [39] C. Marques, "Value proposition as a catalyst for a customer focused innovation," 2010, doi: 10.1108/0025174111183834.
- [40] C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. 2004.
- [41] "MaterialApp Flutter," 2021. <https://api.flutter.dev/flutter/material/MaterialApp-class.html>.
- [42] "Classe ThemeData Flutter," 2021. <https://api.flutter.dev/flutter/material/ThemeData-class.html>.
- [43] "AppBarTheme," 2021. <https://api.flutter.dev/flutter/material/AppBarTheme-class.html>.
- [44] "ButtonThemeData," 2021. <https://api.flutter.dev/flutter/material/ButtonThemeData-class.html>.
- [45] "ChipThemeData," 2021. <https://api.flutter.dev/flutter/material/ChipThemeData-class.html>.
- [46] "ColorScheme," 2021. <https://api.flutter.dev/flutter/material/ColorScheme-class.html>.
- [47] "InputDecorationTheme," 2021. <https://api.flutter.dev/flutter/material/InputDecorationTheme-class.html>.
- [48] "MaterialColor," 2021. <https://api.flutter.dev/flutter/material/MaterialColor-class.html>.

- [49] "AppBarTheme," 2021. <https://api.flutter.dev/flutter/material/AppBarTheme-class.html>.
- [50] "TextStyle," 2021. <https://api.flutter.dev/flutter/painting/TextStyle-class.html>.
- [51] "TextTheme," 2021. <https://api.flutter.dev/flutter/material/TextTheme-class.html>.
- [52] "ToggleButtonsThemeData," 2021.
<https://api.flutter.dev/flutter/material/ToggleButtonsThemeData-class.html>.
- [53] "IconThemeData," 2021. <https://api.flutter.dev/flutter/widgets/IconThemeData-class.html>.
- [54] "MaterialBannerThemeData," 2021.
<https://api.flutter.dev/flutter/material/MaterialBannerThemeData-class.html>.
- [55] "BottomAppBarTheme," 2021.
<https://api.flutter.dev/flutter/material/BottomAppBarTheme-class.html>.
- [56] "BottomNavigationBarThemeData," 2021.
<https://api.flutter.dev/flutter/material/BottomNavigationBarThemeData-class.html>.
- [57] "BottomSheetThemeData," 2021.
<https://api.flutter.dev/flutter/material/BottomSheetThemeData-class.html>.
- [58] "ButtonBarThemeData," 2021.
<https://api.flutter.dev/flutter/material/ButtonBarThemeData-class.html>.
- [59] "CardTheme," 2021. <https://api.flutter.dev/flutter/material/CardTheme-class.html>.
- [60] "NoDefaultCupertinoThemeData," 2021.
<https://api.flutter.dev/flutter/cupertino/NoDefaultCupertinoThemeData-class.html>.
- [61] "DataTableThemeData," 2021.
<https://api.flutter.dev/flutter/material/DataTableThemeData-class.html>.
- [62] "DialogTheme," 2021. <https://api.flutter.dev/flutter/material/DialogTheme-class.html>.
- [63] "DividerThemeData," 2021. <https://api.flutter.dev/flutter/material/DividerThemeData-class.html>.
- [64] "FloatingActionButtonThemeData," 2021.
<https://api.flutter.dev/flutter/material/FloatingActionButtonThemeData-class.html>.
- [65] "NavigationRailThemeData," 2021.
<https://api.flutter.dev/flutter/material/NavigationRailThemeData-class.html>.
- [66] "PopupMenuThemeData," 2021.
<https://api.flutter.dev/flutter/material/PopupMenuThemeData-class.html>.
- [67] "SliderThemeData," 2021. <https://api.flutter.dev/flutter/material/SliderThemeData-class.html>.
- [68] "SnackBarThemeData," 2021.

- <https://api.flutter.dev/flutter/material/SnackBarThemeData-class.html>.
- [69] "TextSelectionThemeData," 2021. <https://api.flutter.dev/flutter/material/TextSelectionThemeData-class.html>.
- [70] "TimePickerThemeData," 2021. <https://api.flutter.dev/flutter/material/TimePickerThemeData-class.html>.
- [71] "TooltipThemeData," 2021. <https://api.flutter.dev/flutter/material/TooltipThemeData-class.html>.
- [72] "Typography," 2021. <https://api.flutter.dev/flutter/material/Typography-class.html>.
- [73] "React GitHub," 2021. <https://github.com/facebook/react>.
- [74] "Flutter GitHub," 2021. <https://github.com/flutter/flutter>.
- [75] "Electron GitHub," 2021. <https://github.com/electron/electron>.
- [76] "React Native GitHub," 2021. <https://github.com/facebook/react-native>.
- [77] "Mobile Frameworks," 2021. <https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>.
- [78] "Web Frameworks," 2021. <https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/>.
- [79] "Electron Apps," 2021. <https://www.electronjs.org/apps>.
- [80] "Flutter," 2021. https://flutter.dev/?gclid=Cj0KCQjw5JSLBhCxARIsAHgO2SepRBoRzcOPasId37_SZpFkf9HvAvZsBkdQx59hTt7hCFQpp7MxYfgaAuS0EALw_wcB&gclidsrc=aw.ds.
- [81] "Apps Flutter," 2021. <https://flutter.dev/showcase>.
- [82] "StatelessWidget Flutter," 2021. <https://api.flutter.dev/flutter/widgets/StatelessWidget-class.html>.
- [83] "StatefulWidget Flutter," 2021. <https://api.flutter.dev/flutter/widgets/StatefulWidget-class.html>.
- [84] "InheritedWidget Flutter," 2021. <https://api.flutter.dev/flutter/widgets/InheritedWidget-class.html>.
- [85] "BuildContext Flutter," 2021. <https://api.flutter.dev/flutter/widgets/BuildContext-class.html>.
- [86] "color_blindness," 2021.
- [87] "FutureBuilder Flutter," 2021. <https://api.flutter.dev/flutter/widgets/FutureBuilder-class.html>.

Anexo A

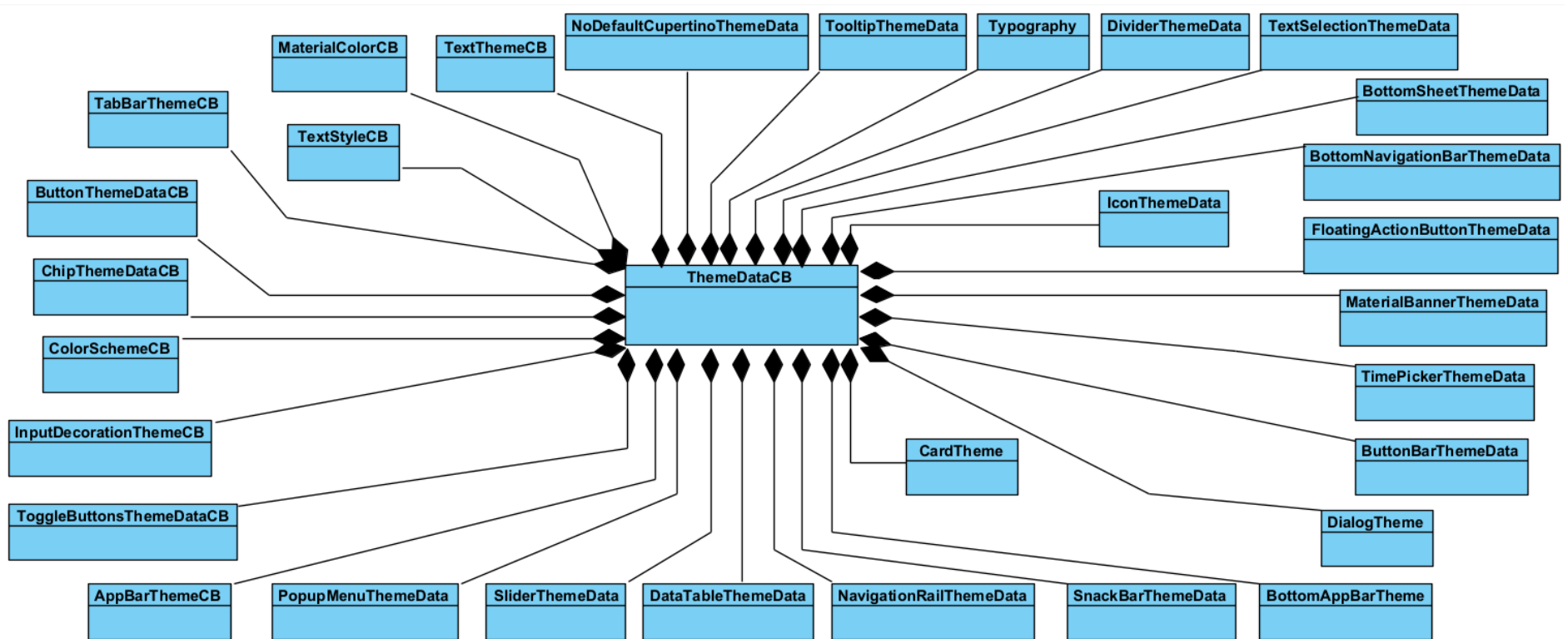


Figura 22 - Diagrama de Classes completo

Anexo B

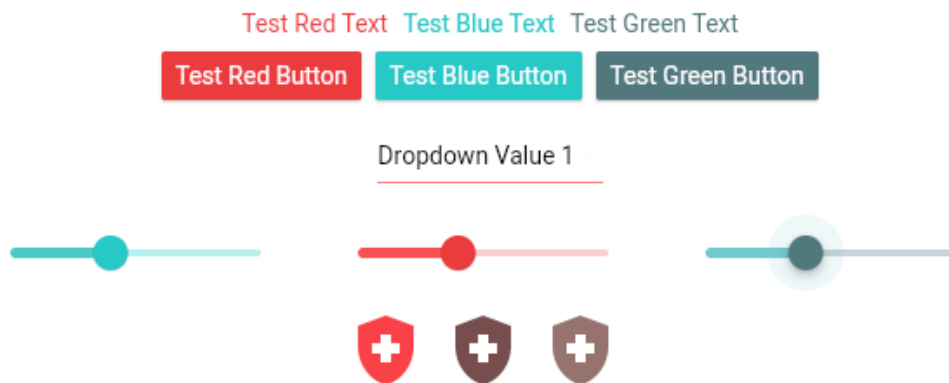


Figura 23 - Aplicação de testes para Tritanopia

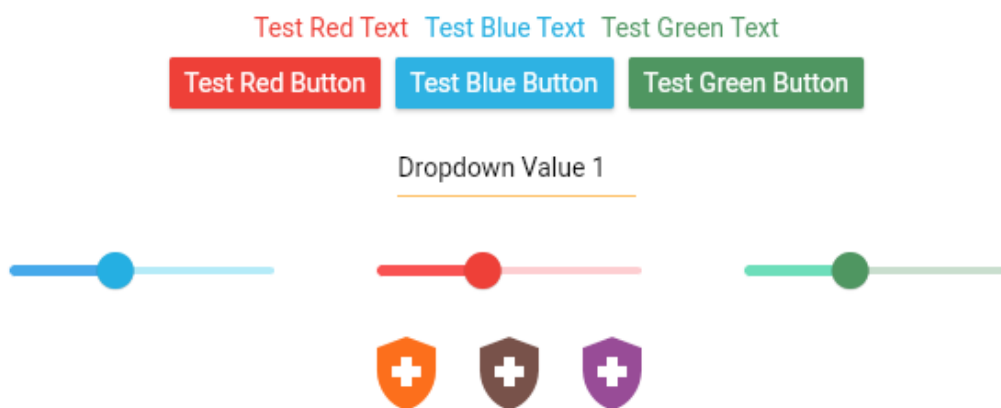


Figura 24 - Aplicação de testes para Tritanomalia

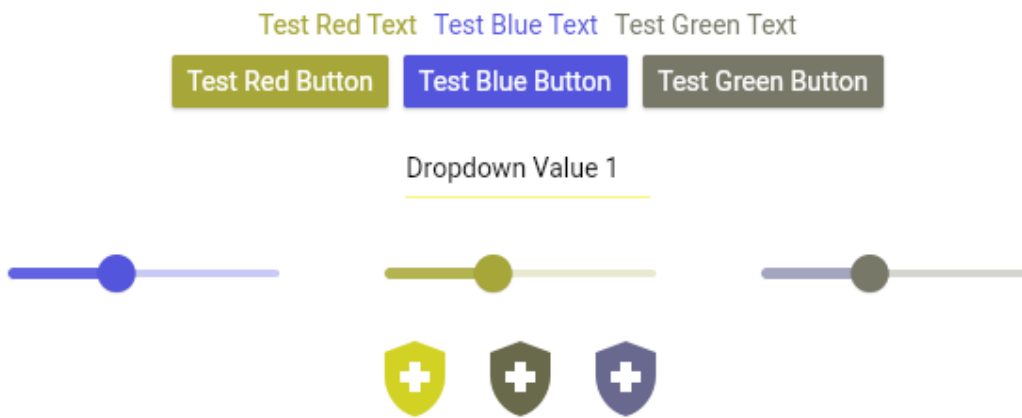


Figura 25 - Aplicação de testes para Protanopia

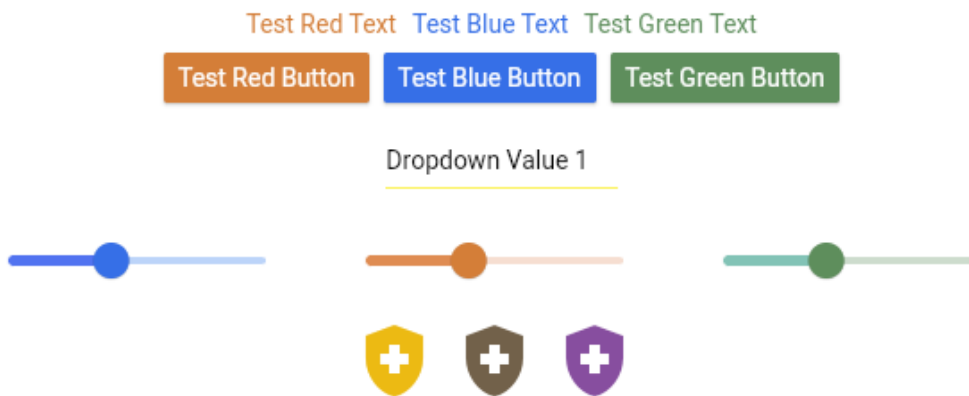


Figura 26 - Aplicação de testes para Protanomalia

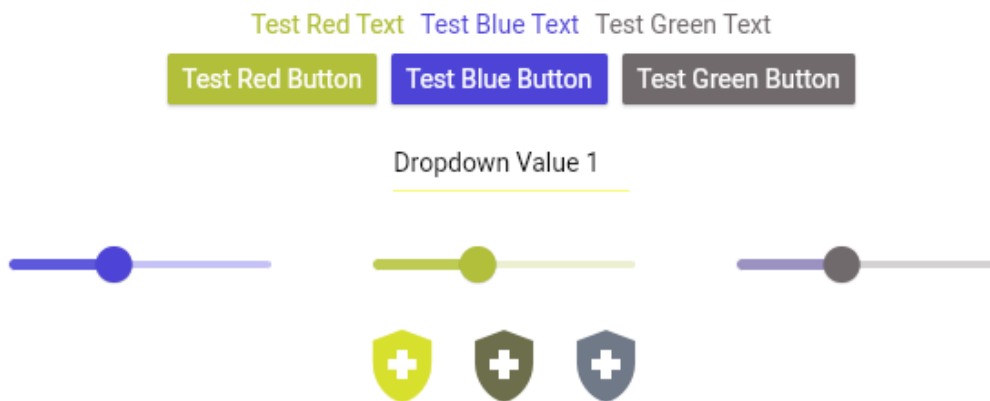


Figura 27 - Aplicação de testes para Deuteranopia

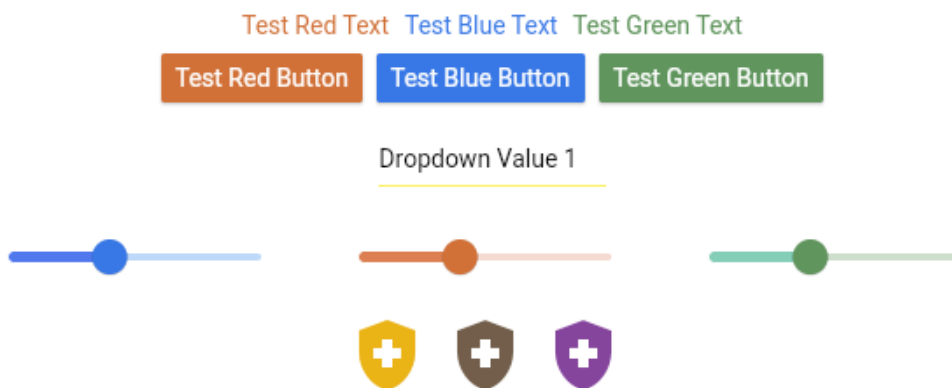


Figura 28 - Aplicação de testes para Deuteranomalia

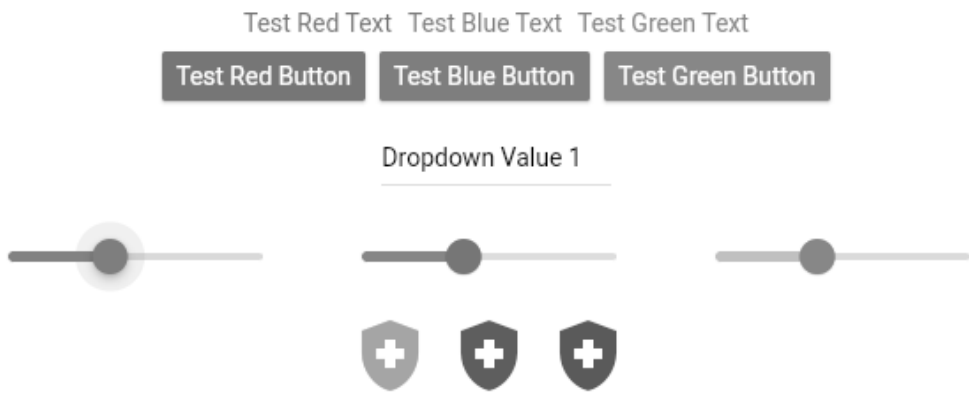


Figura 29 - Aplicação de testes para Monocromacia