

Colorblind Aid: Make Color-Blinded Life Easier

1st Asty Nabilah Izzaturrahmah
dept. of Information System
Hanyang University
Seoul, South Korea
astynabilah@hanyang.ac.kr

2nd Ni Putu Winda Ardiyanti
dept. of Information System
Hanyang University
Seoul, South Korea
windaardiyanti99@gmail.com

Abstract—In this project, a trial is an attempt to make aid for color blinded people so they can detect and classify colors. Color spaces that will be used in this program is RGB (Red Green Blue). The image processing technique is used for identifying the colors in an image. The program has features such as Find A Color, Specify This Color, Articles About Color Blind, Color Blind Test, Mini Quizzes, Find Nearest Eye Doctor, Clothing Guide, and Color Harmonies.

Index Terms—color, colorblind, image processing, RGB color space

I. INTRODUCTION

Color is one of the most important things in human's life. It defines the characteristics of an image. Unfortunately, not all people can see the difference between these colors. The purpose of this project is to develop an application that can solve the color blindness problem, and make colour blinded people can easily see the difference in the colors. The oxford dictionary defined color as "The property possessed by an object of producing different sensations on the eye as a result of the way it reflects or emits light" [1]. A range of colors can be created by the primary colors of pigment and these colors then define a specific color space. Color space, also known as the color model (or color system), is an abstract mathematical model which simply describes the range of colors as tuples of numbers, typically as 3 or 4 values or color components [2]. There are some color spaces, one of them that will be used in this program is RGB (Red Green Blue). An RGB image is a colorful image consisting of fixed values of color contents for each pixel. These color contents have different values ranging from 0 to 255 [3]. Colourblind, also known as color vision deficiency is a decreased ability to see the differences of color. Color blindness may make someone's activity more difficult because color blinded people more difficult to distinguish certain colors such as blue and yellow or red and green [4].

II. FEATURES EXPLANATION

The features of this app are described below.

A. Find A Color

This feature allows user to upload an image and the application will find the color that the user desires. The program will tell which part of the image contains desired color and if user choose 'Specify This Color' then the output will be the color of the picture.

B. Specify This Color

The user will input the image. After that, the image will be processed. An image consists of a lot of pixels (i.e. a physical point in a raster image, or the smallest addressable element in an all points addressable display device [5]). Each pixel corresponds to a code. These codes can be used for the definition of colors and these defined colors are used to recognize colors in the picture after the analyzation.

C. Articles about Color Blind

The program will provides many articles that related to the color blind which user can read, and user can also search articles based on the title.

D. Color Blind Test

The program will display the several tests for the user to take. and after the test is ended, the program will display the result of the test, either the user has colorblind or not, and if so, the program will display the type of colorblind.

E. Mini Quizzes

The program provides multiple-choice questions. Users select one answer and after it, the app will immediately show its right answer and explanation about it.

F. Find Nearest Eye Doctor

This feature is intended to help people to find nearest eye doctor.

G. Clothing Guide

This feature helps people with color blind to choose clothing based on their colors to avoid mismatching colors since they can't distinguish them.

H. Color Harmonies

Color harmonies feature's way of work is not quite different with clothing guide. The difference is, color harmonies only shows tonal color while the clothing guide shows not only tonal colors but also main colors and some non-tonal color that still match with it.

III. SIMILAR PROJECTS

A. Paper titled 'Color palette extraction with K-means clustering: Machine Learning from Scratch' by Nandini Bansal; this project is talk about ho to extract the color using K-means Clustering, we planning to use the same method as this project to develop our software project

B. App called 'Color Grab (color detection)' by Loomatix Team

There are some similar feature from Color Grab App with our software project, which are *Find Harmonies*; where user can find the perfect color combinations, *Capture*; to recognize and pick the color from a picture, and *Photos*; grab colors from a picture

C. Aplikasi Tes Buta Warna dengan Metode Ishihara Pada Smartphone Android (Colorblind Test App using Ishihara Method in Android Smartphone) by Randy Viyata Dhika, Ernawati, and Desi Andreswari

IV. DEVELOPMENT ENVIRONMENT

A. Web Platform

We decided to use web platform in building this software because our team have more experience in building a website. Website also allows the user easier to access our software in website.

B. Programming Languages

- JavaScript version 1.8.5, a dynamic computer programming language which commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.
- Hypertext Markup Language revision 5 (HTML5) is markup language for the structure and presentation of World Wide Web contents. HTML5 supports the traditional HTML and XHTML-style syntax and other new features in its markup, New APIs, XHTML and error handling.

C. Frameworks

- CodeIgniter ver 3.1.11, a kind of MVC (Model View Controller) framework.
- Bootstrap ver 4.4.1, a front-end framework, used to build user interfaces. It is CSS (Cascading Style Sheet) and Javascript based-design templates.

D. API in use

- Google API (Maps JavaScript API) to run the 'Find The Nearest Eye Doctor' feature. The cost of Google Map API is 7 USD for 1000 cost.

E. Software in Use

- Visual Studio Code ver 1.38.1, a text editor app which used to code.
- XAMPP ver 2.4.38.0, a cross-platform web server solution stack package.
- phpMyAdmin ver 4.9.2, a software which used to handle administration of SQL over web.
- Atom ver. 1.41.0, a text editor app which used to code.

F. OS Used

Windows 10 is used since it's easy and familiar for us.

G. Cost Estimation

For the feature called 'Find The Nearest Eye Doctor', Google Map API will be used and the price is 7 USD for 1000 requests. Since we only will do 1 request, the API will be free. So, 7 USD will be our maximum expenses.

H. Task Distribution

- Asty : Web back-end, front-end, Documentation, Managing the project
- Winda : Web back-end, front-end, Documentation, Managing the project

V. SPECIFICATIONS

A. Find A Color

First, we need to define the upper and lower limits for pixel values based on the color that we want find. Then we look into data set and specifying which pixels fall into specified upper and lower range. Then it will show image with only one color that was intended to be found.

B. Specify This Color

The method that will be used is K-Means Clustering. K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K [6]. First, we convert the image to points that our clustering algorithm can use. Next, the color distance is calculated using Euclidean distance formula, which is:

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (1)$$

After that, we find the center for a set of points by adding the values for each dimension and divide by the number of points. Then, the clusters are sorted and the value will be converted into hexadecimal form.

C. Articles about Color Blind

The articles will be found manually and then inputted to the app along with the article source (e.g. URLs). Admin will type the article in the text box provided and the program will get the value from the input in HTML. After that, the article will be posted.

D. Color Blind Test

First, the web will shows 25 pictures from *Ishihara Plate*. The user will input what number do they see from each picture. After the submit button is clicked, the program will calculate the right answer. If the wrong answer is more than half of total pictures, the result will be colorblinded, if half of total pictures then the result is partially colorblinded, and normal otherwise.

E. Mini Quizzes

The flow is actually quite similar with colorblind test, the difference is, the quiz will give the right answer right after the user input their answer.

F. Find Nearest Eye Doctor

we decide to purchase the Google Map API to create this feature, the step to build this feature is: a. Creating a table in MySQL, in this step we creating a table that containing attributes of the markers on the map, like the marker id, name, address, lat, lng. To keep the storage space for the table is minimum, we specify the lat and lng attributes to be floats of size(10,6). this allows the fields to store 6 digits after the decimal and plus up to 4 digits before the decimal. b. Populating the table, the next step is we input the data to the database table that we create. c. Outputting data as XML using PHP, in this step we should have a table named markers that containing the map marker data. this section show us how to export the table data from the SQL database in an XML format. the map can use the XML file to retrieve the marker data through asynchronous JavaScript calls. Using an XML file as an intermediary between our database and our Google map allows for faster initial page load, and a more flexible map application. It makes debugging easier as we can independently verify the XML output from the database, and the JavaScript parsing of the XML. we also can run the map entirely based on static XML files only, and not use the MySQL database. d. Finding locations with MySQL, to find locations in our markers table that are within a certain radius distance of a given latitude/longitude, we use a SELECT statement based on the Haversine formula. The Haversine formula is used generally for computing great-circle distances between two pairs of coordinates on a sphere.

e. Using PHP's DOM XML functions to output XML, the DOM XML functions of PHP take care of subtleties such as escaping special entities in the XML, and make it easy to create XML with more complex structures. we use DOM XML functions to create XML nodes, append child nodes, and output an XML document to the screen. f. Creating the map, This section shows how to develop the map example using JavaScript, and the output XML file. first thing that we do is setting up the controls, we have to sets up the following controls on the map such as 'Search Near user input field', 'Radius drop down list', 'Search button', 'See all results drop down list of search results'. and after that we pulling it all together all the steps.

G. Clothing Guide

The idea of this feature is to find the matching color of the clothes, the color will be matching if the color has the same tone or the distance tone between two color and another is quite similar. To apply this feature, first is we convert the color from RGB form to hexadecimal. since the hexadecimal has 256 character so the data of that we receiving is from 0 to 255. After we already have the color in hexadecimal, we will calculate the distance between the two color, we will use the euclidean distance formulas to calculate the distance of the color formulas : the result of the algorithm will be give us

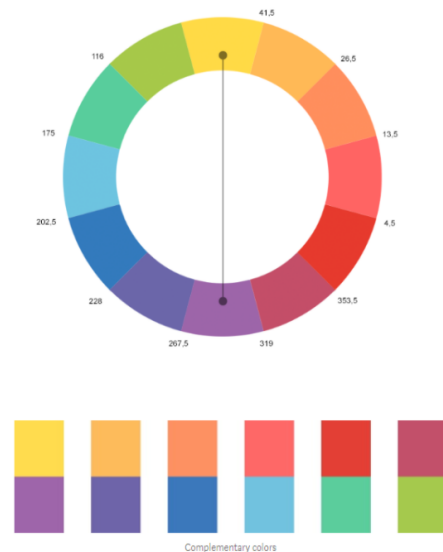
$$\text{distance} = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2}$$

the distance value between two colors, if the distance is small then the color is match, but if the distance is large the color is doesn't match.

H. Color Harmonies

To run this feature, there are a several steps to build the algorithm for this feature. The first step is determine the average color of the image, we have to reducing the image to make it easy to determine the average of the color, and after that we convert the average color to the RGB form and from RGB form to hexadecimal form. The second step is defining its complementary color in the color wheel (we create the function that create the color wheels that has parameter tones/hue(ranges 0-360°). Next, we calculate the area of the color objects to determine the saturation of the complementary color. Since the area color objects differs, it is necessary to equalize them by changing the saturation or lightness of color.

- Determine the image area and substrate.
- Calculating the coefficient (substance divided by image)
- Set the relationship of lightness of basic complementary colors, we already set the lightness of basic complementary colors with ratio: Yellow: Purple = 9 : 3 = $\frac{3}{4}$: $\frac{1}{4}$



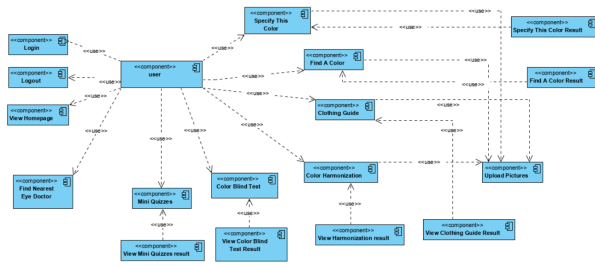


Fig. 1. Component Diagram of This Program

Orange: Blue = $8 : 4 = 2/3 : 1/3$ Red : Green = $6 : 6 = 1 : 1 = 1/2 : 1/2$ AVERAGE : 2 : 1 Find Y, where Y is the result of coefficient divide with the average After that find X, where $X = 100\text{percent} : Y$ Last, we reduce the lightness of the substance by overlaying the white substrate with n percent opacity ($100\text{percent} - X$)

- The output image will be the result after we reduce the lightness substance.

VI. ARCHITECTURE DESIGN AND IMPLEMENTATION

A. Architectural Description

Architectural Description is a description of the architecture or component that will be applied to the software ‘Colorblind Aid: Make Color-Blinded Life Easier’ to facilitate developers in developing or implementing this software.

B. Component Description

No	Component's Name	Explanation
1	User	User of Colorbuddy Software
2	View Homepage	Menu to view the homepage
3	Find Nearest Eye Doctor	Menu to find the nearest eye doctor in specific areas
4	Articles of Color Blind	Menu that contains articles that related with color blind
5	Color Blind Test	Menu that contains a test to check if user is colorblinded or not.
6	Mini Quizzes	Menu that provide the quizzes about color blind, to improve the knowledge about color blind
7	Clothing Guide	Menu to give the user clothing guide advice, how to mix and match a color in choose the cloths that user want to wear
8	Color Harmonization	Menu that will give the user suggestion of harmonization color, when user want to match a color to do some purpose (e.g.: photo editing, designing)
9	Specify this Color	Menu to specify the colors available at the uploaded picture
10	Find A Color	Menu to find a current color on the picture that user uploaded
11	View Harmonization Result	Menu to view the result of the ‘Color Harmonization’ menu
12	View Clothing Guide Result	Menu to view the result of the ‘Clothing Guide’ menu
13	Specify This Color Result	Menu to view the result of ‘Specify This Color’ menu
14	Find A Color Result	Menu to view the result of ‘Find A Color’ menu
15	Upload Picture	Menu to upload the image

C. Class Identification

- Find Nearest Eye Doctor

No	Design Class Name	Class Type
1	User	Actor
2	Find Nearest Doctor Page	Boundary
3	Find Nearest Doctor Controller	Controller
4	Location	Database

- View Homepage

No	Design Class Name	Class Type
1	User	Actor
2	Homepage	Boundary
3	View Homepage Controller	Controller

- Articles Color Blind

No	Design Class Name	Class Type
1	User	Actor
2	Articles Colorblind Page	Boundary
3	Article Colorblind Controller	Controller
4	Colorblind Articles	Database

- Color Blind Test

No	Design Class Name	Class Type
1	User	Actor
2	Colorblind Test Page	Boundary
3	Colorblind Test Controller	Controller
4	Colorblind Test Answers	Database

- Mini Quizzes

No	Design Class Name	Class Type
1	User	Actor
2	Mini Quizzes Page	Boundary
3	Mini Quizzes Controller	Controller
4	Mini Quizzes Answers	Database

- Clothing Guide

No	Design Class Name	Class Type
1	User	Actor
2	Clothing Guide Page	Boundary
3	Clothing Guide Controller	Controller
4	Clothing Guide Database	Database

- Color Harmonization

No	Design Class Name	Class Type
1	User	Actor
2	Color Harmonization Page	Boundary
3	Color Harmonization Controller	Controller
4	Color Harmonization Database	Database

- Specify This Color

No	Design Class Name	Class Type
1	User	Actor
2	Specify This Color Page	Boundary
3	Specify This Color Controller	Controller
4	Specify This Color Database	Database

- Find A Color

No	Design Class Name	Class Type
1	User	Actor
2	Find A Color Page	Boundary
3	Find A Color Controller	Controller
4	Find A Color Database	Database

- Upload Image

No	Design Class Name	Class Type
1	User	Actor
2	Upload Image Page	Boundary
3	Upload Image Controller	Controller
4	Image	Database

- View Harmonization Result

No	Design Class Name	Class Type
1	User	Actor
2	View Harmonization Result Page	Boundary
3	View Harmonization Result Controller	Controller
4	Color Harmonization	Database

- View Clothing Guide Result

No	Design Class Name	Class Type
1	User	Actor
2	View Clothing Guide Result Page	Boundary
3	View Clothing Guide Result Controller	Controller
4	Clothing Guide Database	Database

- Specify This Color Result

No	Design Class Name	Class Type
1	User	Actor
2	Specify This Color Result Page	Boundary
3	Specify This Color Result Controller	Controller
4	Specify This Color Database	Database

- Find A Color Result

No	Design Class Name	Class Type
1	User	Actor
2	Find A Color Result Page	Boundary
3	Find A Color Result Controller	Controller
4	Find A Color Database	Database

VII. MOCKUP DESIGNS

Below are some mockup designs that is planned to be implemented in the program.



Fig. 2. Mockup for View Article Page

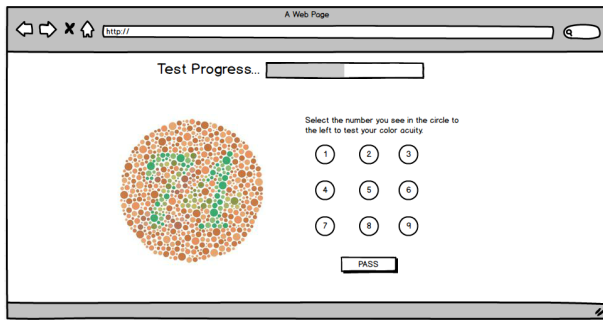


Fig. 3. Mockup for Colorblind Test Page



Fig. 6. Mockup for Clothing Guide Page

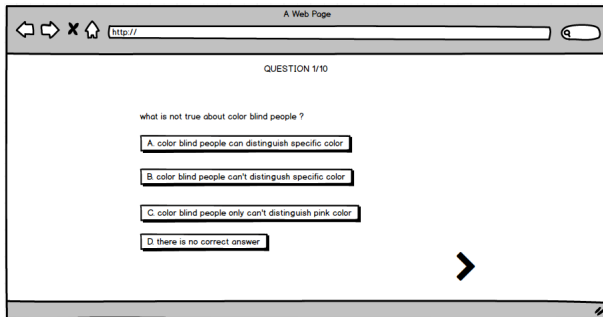


Fig. 4. Mockup for Mini Quiz Page

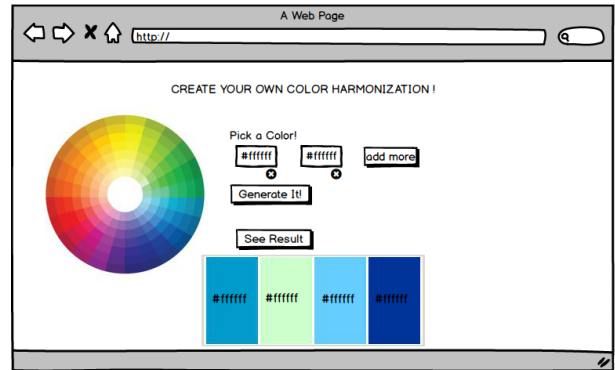


Fig. 7. Mockup for Color Harmonization Page

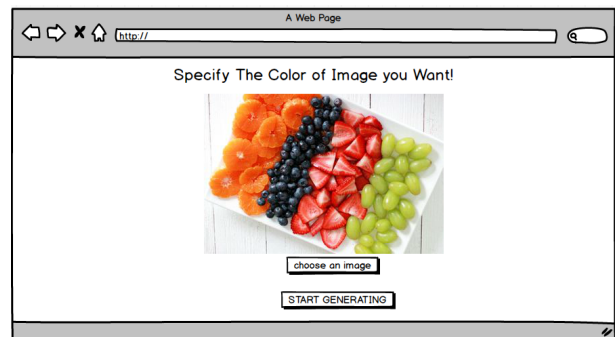


Fig. 8. Mockup for Specify This Color Page

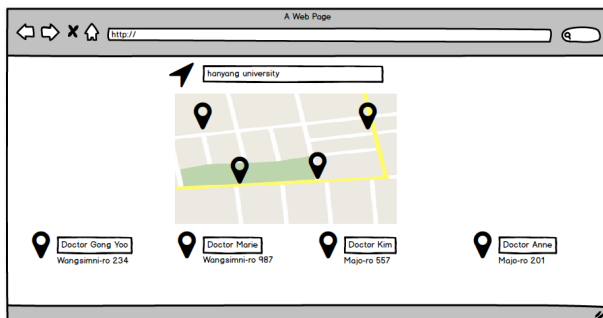


Fig. 5. Mockup for Find Nearest Eye Doctor Page

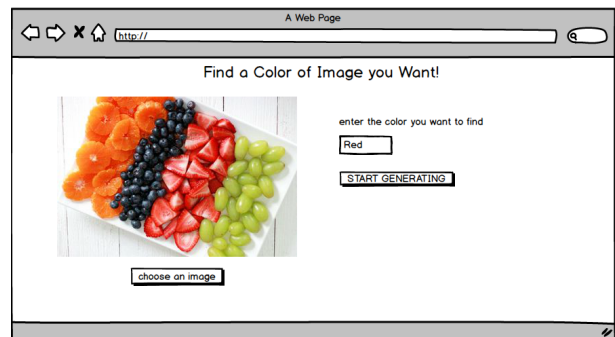


Fig. 9. Mockup for Find A Color Page

VIII. DIRECTORY ORGANIZATION

Below are file directories in our project.

Directory	File Name	Module Name in Use
/ColorBlindBuddy/ application/views/	home _V .php	View Module
/ColorBlindBuddy/ application/views/	articles _V .php	View Module
/ColorBlindBuddy/ application/views/	harmonization _V .php	View Module
/ColorBlindBuddy/ application/views/	test _V .php	View Module
/ColorBlindBuddy/ application/views/	testresult _V .php	View Module
/ColorBlindBuddy/ application/views/	quiz _V .php	View Module
/ColorBlindBuddy/ application/views/	quizresult.php	View Module
/ColorBlindBuddy/ application/views/	cGuide _V .php	View Module
/ColorBlindBuddy/ application/views/	findColor _V .php	View Module
/ColorBlindBuddy/ application/views/	specifyColor _V .php	View Module
/ColorBlindBuddy/ application/views/	findDolor _V .php	View Module
/ColorBlindBuddy/ application/views/	header.php	View Module
/ColorBlindBuddy/ applica- tion/controllers	home.php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	articles _C .php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	findcolor _C .php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	clothingguide.php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	Colorblindtest.php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	eyedoctor.php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	harmonies.php	Controller Module
/ColorBlindBuddy/ applica- tion/controllers	quiz.php	Controller Module

/ColorBlindBuddy/ applica- tion/controllers	specifycolor _C .php	Controller Module
/ColorBlindBuddy/ application/model	quizmodel.php	Model Mod- ule
/ColorBlindBuddy/ application/model	testmodel.php	Model Mod- ule
/ColorBlindBuddy/ application/database	colorbuddy.sql	Database Module

A. View Module

We create the software front-end design on this module, view module contains the page design that create separately for every page that consist on our website. we create the view module to make us easier to distinguish the front-end and back-end part when we making the website.

B. Controller Module

Controller module have a role to connect view with model, and also set the display of the website page.

C. Model Module

Model module have a role to manages the data, logic and rules of the website. For example if we create the page that contains the command to load the database, so we execute the algorithm on model module.

D. Database Module

this module is contains the database that use on this project, we are using the database to develop the quiz page and color blind test page.

IX. USE CASES

A. Find A Color

- Name: Find A Color
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor click 'find a color' page	
	System shows the page
Actor input image and the name of color that they wants to find	
	System generates the input and then execute it then shows the color that actor wants to find

- Use Case Diagram

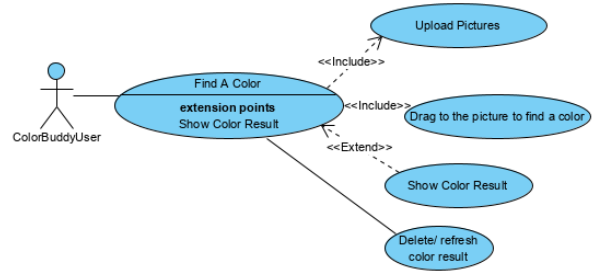


Fig. 10. Find A Color Use Case Diagram

B. Specify This Color

- Name: Specify This Color
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor click 'Specify the Color' page	
System shows the page	
Actor gives input the image	
	System reads the image and execute the color on the image and shows the result

- Use Case Diagram

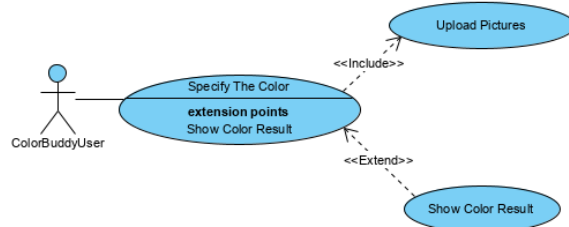


Fig. 11. Specify This Color Use Case Diagram

C. Colorblind Articles

- Name: Colorblind Articles
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor click 'article' page	
	System shows the article page

- Use Case Diagram

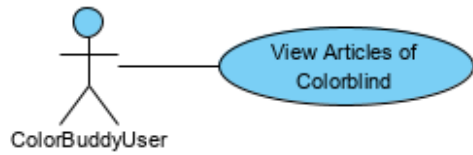


Fig. 12. Colorblind Article Use Case Diagram

D. Colorblind Test

- Name: Colorblind Test
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor clicks 'color blind test' page	
	System shows the color blind test page
Actor gives answer on the test	
	System read actor's answer and generate the result of the test then shows the color blind test result page

- Use Case Diagram

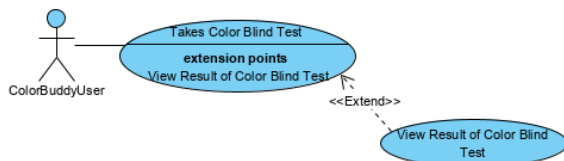


Fig. 13. Colorblind Test Use Case Diagram

E. Mini Quizzes

- Name: Mini Quizzes
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor clicks 'mini quizzes' pages	
	System shows the mini quizzes page
Actor fills out the answer of the quiz	
	System reads actor's answer and generates and shows the result of the quiz

- Use Case Diagram

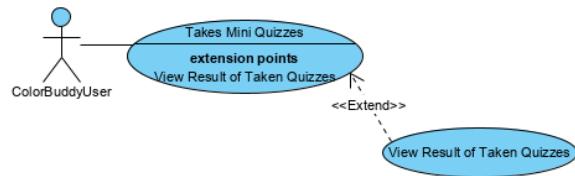


Fig. 14. Quiz Use Case Diagram

F. Find Nearest Eye Doctor

- Name: Find Nearest Eye Doctor
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor click 'find the nearest eye doctor' menu	
	System tracks user's location and use google maps API to find the nearest eye doctor from actor location and shows the nearest eye doctor list

- Use Case Diagram

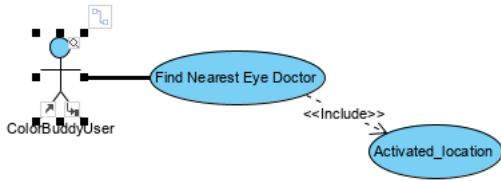


Fig. 15. Find Nearest Eye Doctor Use Case Diagram

G. Clothing Guide

- Name: Clothing Guide
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor click 'find clothing guide' pages	
	System shows 'find clothing guide' page
Actor gives the image of the clothes that he/she want to match with	
	System generates input and shows the result

- Use Case Diagram

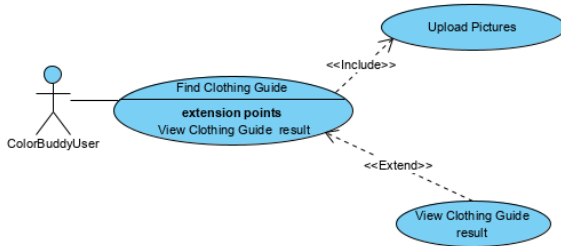


Fig. 16. Clothing Guide Use Case Diagram

H. Color Harmonies

- Name: Color Harmonies
- Actor: ColorBuddy User
- Flow of Events:

Actor	System
Actor click 'check the color harmonies' page	
	System shows the page
Actor picks color	
	System generates the color, finds the tonal colors, then shows the result

- Use Case Diagram

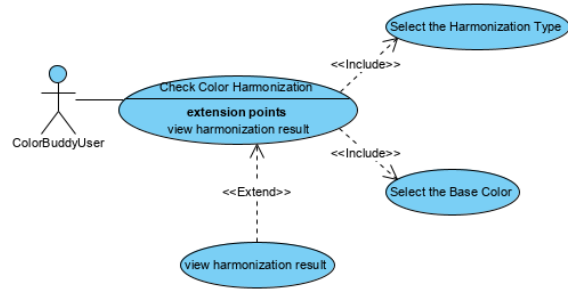


Fig. 17. Color Harmonies Use Case Diagram

X. DISCUSSION

On this final project, we decided to create a project for color blind people after eliminating a lot of any other idea. Our project team's members are just 2 persons with no great experience, so there is a little struggle when creating this website. Some problems including running out the time, lack of experience, and confusion about how to build the website so it can meet our requirements, that was the hardest part. There's also imperfections in our functions and more things to improve. There's also plan changing to adjust time and skills that we have. Some interfaces and algorithms are different from the plan in order to gain simplicity during limited time. In Colorblind test, we need to improve more details on what type of colorblind that users have. In Mini Quizzes, we can add so much more questions and give more nice interface to make it looks fun and educational as it is planned to be. This is also our first time to compile a \LaTeX document and that's why there's some imperfections in this document too. Finally, we can create this software and document, even though there are several things that does not meet our expectations. There are a lot of things that we can learn when we make this software, such as how to manage time, how to make the good document for the software requirements.

REFERENCES

- [1] <https://www.lexico.com/en/definition/colour>
- [2] <http://www.arcsoft.com/topics/photostudio-darkroom/what-is-color-space.html>
- [3] Arshi prabhakar, Neetiand Rakhi Devi.2017. "Different Color Detection in an RGB Image", International Journal of Development Research, 7,(08).
- [4] <https://www.allaboutvision.com/conditions/colordeficiency.htm>
- [5] Rudolf F. Graf (1999). Modern Dictionary of Electronics. Oxford: Newnes. p. 569.
- [6] <https://blogs.oracle.com/datascience/introduction-to-k-means-clustering>

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove the template text from your paper may result in your paper not being published.