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 color 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]. Colorblind, 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 provide 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 is:

- *Find Harmonies; where this feature is to find the perfect color combinations*
- *Capture; this feature is to recognize and pick the color from a picture*
- *Photos; this feature to 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. Choice of Software Development Platform

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) Programing Language

- JavaScript version 1.8.5
  JavaScript is a dynamic computer programming language. It is lightweight and most 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.
- HTML5
  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) Cost Estimation

- Google Map JavaScript API Cost
  the cost of Google Map API is 7 USD for 1000 cost. Since we only will do 1 request, the API will be free. So, 7 USD is our maximum expenses.

B. Software in use

- Google API (Maps JavaScript API)
  We are using this software to run the feature 'Find the Nearest Eye Doctor'
- K-Means Clustering.

V. SPECIFICATIONS

A. Register

User have to register their account first before they can login to the application. When register, user must enter their username, gender, date of birth, email address and password. After user enter the register requirements the system will check the data from user whether already in database or not, if the data already in the database, then the system will send a message to user that the account already in used and user only have to login, if not then the system will input the data to the database and send a message to the user that the register success.

B. Login

User have to login first before they use the application. for login, user must input the username and the password that they already registered. and after user input the data, system will check whether the data already at the database or not. if the data already at the database, then the system will send a pop-up message that the login is success, if not then the system will send the pop-up message about user caused login failure.

C. 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.

D. 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:

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

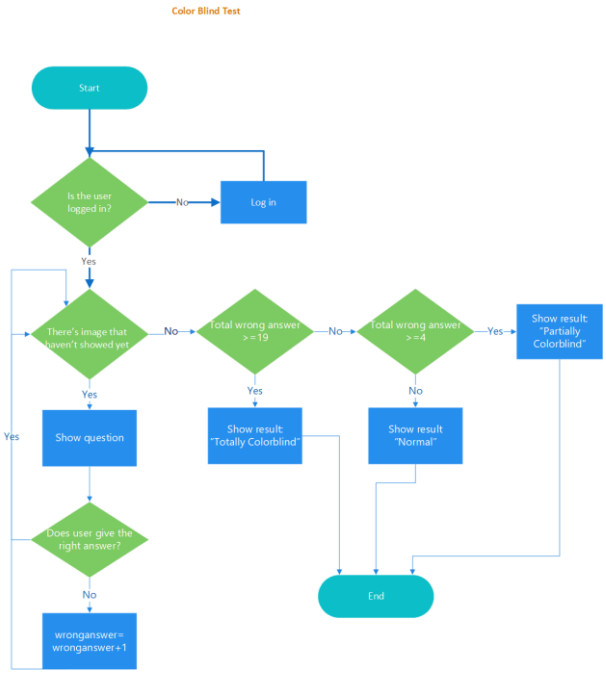
After that, we find the center for a set of points by adding the values for each dimension and divide by the number of points. Then, the clusters are sorted, and the value will be converted into hexadecimal form.

E. Articles about Color Blind

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

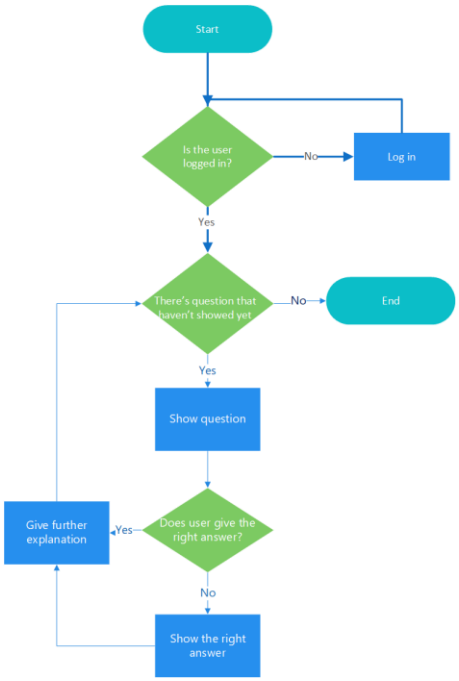
F. Color Blind Test

The flow for the algorithm descripted in image below:



G. Mini Quizzes

Explained in flowchart:



H. 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 are 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 shows 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 XLM*, 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 are pulling it all together all the steps.

I. 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 :

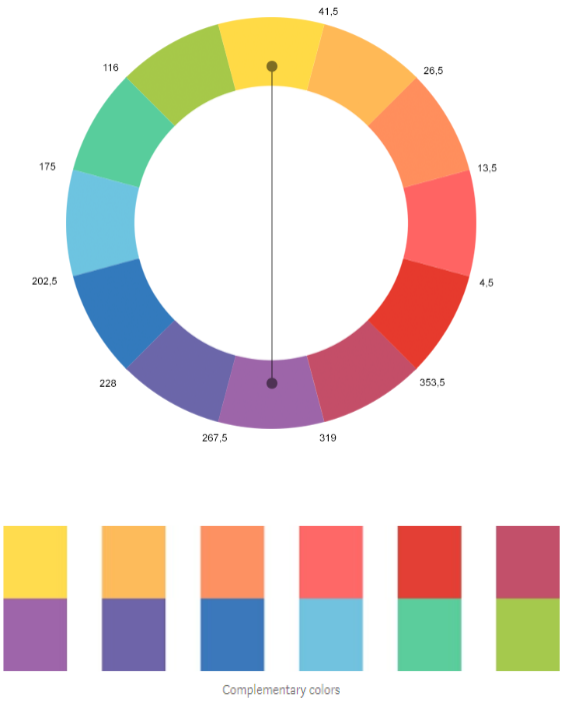
$$\text{distance} = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2}$$

the result of the algorithm will be giving us 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.

J. Color Harmonies

To run this feature, there are a several steps to build the algorithm for this feature.

- The first step is determining 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 after we determine the average color on the image, we define its complementary color (harmonious) of the average color in the color wheel (we create the function that create the color wheels that has parameter tones/hue(ranges 0-360°)



- The next step is 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 = ¾ : ¼
    Orange: Blue = 8 : 4 =2/3 : 1/3
    Red : Green = 6 : 6 = 1:1 = ½ : ½
    AVERAGE : 2 : 1
  - Find Y, where Y is the result of coefficient divide with the average
  - After that find X, where X = 100% : Y
  - Last, we reduce the lightness of the substance by overlaying the white substrate with n% opacity (100% - X)

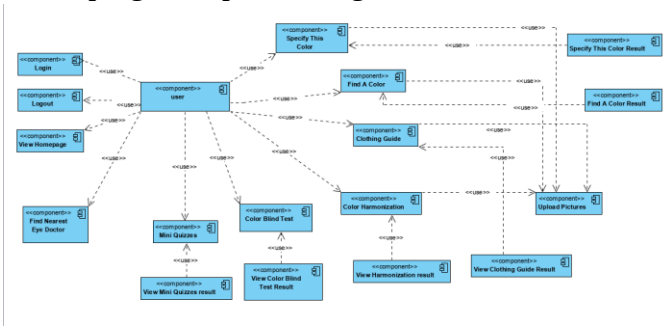
The output image will be the result after we reduce the lightness substance.

V. ARCHITECTURE DESIGN & IMPLEMENTATION

A. Overall Architecture

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	Someone that become the user of 'Colorblind Air' software
2	Create User Account	Menu to create the user account before login to the software
3	View Homepage	Menu to view the homepage
4	Login	Menu to login to the application
5	Logout	Menu to logout from the application
6	Find Nearest Eye Doctor	Menu to find the nearest eye doctor in specific areas
7	Articles of Color Blind	Menu that contains articles that related with color blind issues
8	Color Blind Test	Menu that contains a test to diagnose the people that they have a color blind or not, to know the type of the color blind that people may have
9	Mini Quizzes	Menu that provide the small quizzes about color blind issues, to improve the knowledge about color blind
10	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
11	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 editor, design)
12	Specify this Color	Menu to specify the color at the picture that user uploaded
13	Find A Color	Menu to find a current color on the picture that user uploaded
14	View Mini Quizzes Result	Menu to view the result of the 'Mini Quizzes' menu
15	View Color Blind Test Result	Menu to view the result of the 'Color Blind Test' menu
16	View Harmonization Result	Menu to view the result of the 'Color Harmonization' menu
17	View Clothing Guide Result	Menu to view the result of the 'Clothing Guide' menu
18	Specify This Color Result	Menu to view the result of 'Specify This Color' menu
19	Find A Color Result	Menu to view the result of 'Find A Color' menu
20	Upload Picture	Menu to upload the image

c). Class Identification and Class Diagram

- Create User Account

a) Class Identification

No	Design Class Name	Class Type
1	User	Actor
2	Create User Account Page	Boundary
3	Create User Account Controller	Controller
4	User Database	Database

b) Class Diagram

- User Login

a) Class Identification

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

b) Class Diagram

- User Logout

a) Class Identification

No	Design Class Name	Class Type
1	User	Actor
2	Logout Form	Form
3	Logout Controller	Controller

b) Class Diagram

- Find Nearest Eye Doctor

a) Class Identification

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

b) Class Diagram

- View Homepage

a) Class Identification

No	Design Class Name	Class Type
1	User	Actor
2	Homepage	Boundary
3	View Homepage Controller	Controller
4	(what will we have showed on the homepage)	Database

b) Class Diagram

- Articles Color Blind

a) Class Identification

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

b) Class Diagram

- Color Blind test

a) Class Identification

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

b) Class Diagram

- Mini Quizzes

a) Class Identification

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

b) Class Diagram

- View Result

a) Class Identification

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

4	Result	Database

b) Class Diagram

- Clothing Guide
  a) Class identification

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

b) Class Diagram

- Color Harmonization
  a) Class Identification

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

b) Class Diagram

- Specify This Color
  a) Class Identification

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

b) Class Diagram

- Find A Color
  a) Class Identification

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

b) Class Diagram

- Upload Image
  a) Class Identification

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

b) Class Diagram

- View Mini Quizzes Result
  a) Class Identification

No	Design Class Name	Class Type
1	User	Actor
2	View Mini Quizzes Result Page	Boundary
3	View Mini Quizzes Result Controller	Controller

4	Mini Quizzes Result	Database

b) Class Diagram

- View Color Blind Test Result
  a) Class Identification

No	Design Class name	Class Type
1	User	Actor
2	View Color Blind Test Result Page	Boundary
3	View Color Blind Test Result Controller	Controller
4	Color Blind Test Result	Database

b) Class Diagram

- View Harmonization Result
  a) Class Identification

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

b) Class Diagram

- Use Case View Clothing Guide Result
  a) Class Identification

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

b) Class Diagram

- Specify This Color Result
  a) Class Identification

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

b) Class Diagram

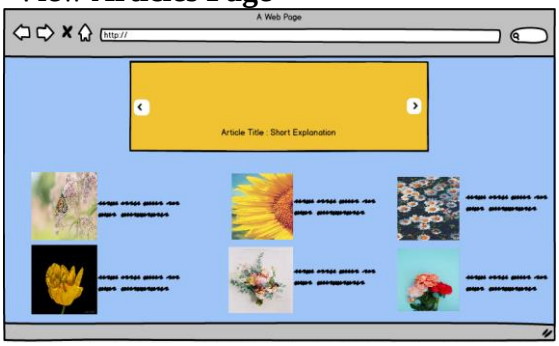
- Find A Color Result
  a) Class Identification

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

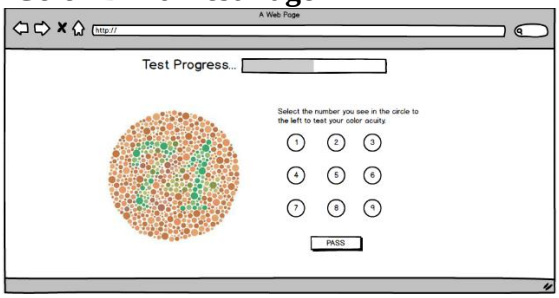
b) Class Diagram

d). Mockups Design

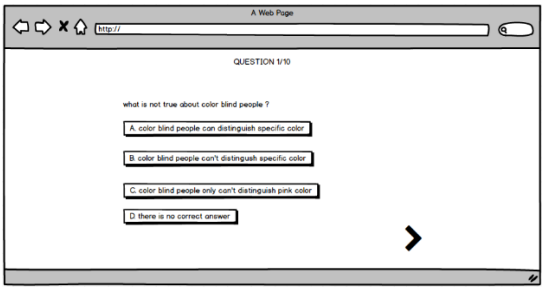
❖ View Articles Page



❖ Color Blind Test Page



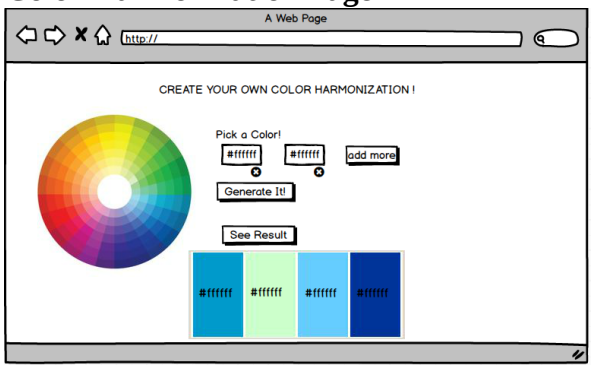
❖ Mini Quizzes Page



❖ Find Nearest Eye Doctor Page

❖ Clothing Guide Page

❖ Color Harmonization Page



❖ Specify This Color Page

❖ Find A Color Page

B. Directory Organization

Directory	File Names	Module Names in Use
/ColorBlindBuddy/application/views/	home_V.php	View Homepage
/ColorBlindBuddy/application/views/	articles_V.php	View Articles page
/ColorBlindBuddy/application/views/	harmonization_V.php	Check Color Harmonization page
/ColorBlindBuddy/application/views/	cbTest_V.php	Color Blind test page
/ColorBlindBuddy/application/views/	mQuiz_V.php	Mini Quizzes page
/ColorBlindBuddy/application/views/	cGuide_V.php	Clothing Guide page
/ColorBlindBuddy/application/views/	findColor_V.php	Find a Color page
/ColorBlindBuddy/application/views/	specifyColor_V.php	Specify This Color page
/ColorBlindBuddy/application/views/	findDoctor_V.php	Find the Nearest Eye Doctor page
/ColorBlindBuddy/application/views/	login_V.php	Login Page
/ColorBlindBuddy/application/views/	logout_V.php	Logout Page
/ColorBlindBuddy/application/views/	header.php	-
/ColorBlindBuddy/application/views/	footer.php	-
/ColorBlindBuddy/application/controllers	Home_c.php	View Homepage Controller
/ColorBlindBuddy/application/controllers	Articles_c.php	Articles Controller
/ColorBlindBuddy/application/controllers	Harmonization_c.php	Check Color Harmonization Controller
/ColorBlindBuddy/application/controllers	cbTes_c.php	ColorBlind Test Controller
/ColorBlindBuddy/application/controllers	mQuiz_c.php	Mini Quizzes Controller
/ColorBlindBuddy/application/controllers	cGuide_c.php	Clothing Guide Controller

/ColorBlindBuddy/application/controllers	findColor_c.php	Find a Color Controller
/ColorBlindBuddy/application/controllers	specifyColor_c.php	Specify this Color Controller
/ColorBlindBuddy/application/controllers	findDoctor_c.php	Find the Nearest Eye Doctor Controller
/ColorBlindBuddy/application/controllers	login_c.php	Login Controller
/ColorBlindBuddy/application/controllers	logout_c.php	Logout Controller
/ColorBlindBuddy/application/models	project_model.php	Project Models
/ColorBlindBuddy/system/database/ColorBuddy/	ColorBuddy.sql	ColorBuddy database

C.

VI. USE CASES

A. USE CASE LOGIN

a. Name of Use Case: Login

b. Actor: ColorBuddy User

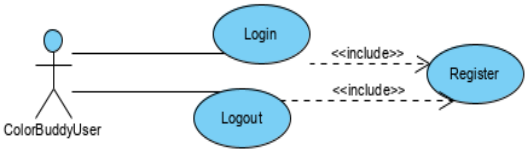
c. Flow of Events

Actor	System
Click 'Login' Button	
	Display the login page
Enter Username and Pasword	
	System will check whether the username and password match with data on database
	If data is correct, system will send the pop-up message to actor that login is success
	System will display the homepage after login

Exceptional:

	Username and password are incorrect
	System will send the message popup to actor that username or password didn't match
Actor will re-input the password or username	

d. Use Case Diagram



B. USE CASE LOGOUT

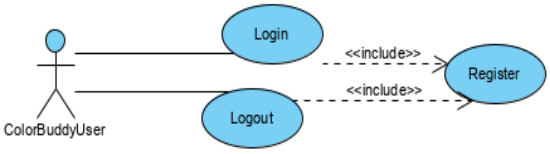
a. Name of Use Case: Logout

b. Actor: ColorBuddy User

c. Flow of Events

Actor	System
Actor click ‘logout’ button	
	System will end the login session of the actor

d. Use Case Diagram



C. USE CASE REGISTER

a. Name of Use Case: Register
b. Actor: ColorBuddy User
c. Flow of events

Actor	System
Actor click the ‘register’ button	
	System will display the register form page
Actor fill out the register requirements	
	System will read the data that input by user and check on the database, whether the data is already registered or not
	If not, system will save actor’s data, and create the login account for actor
	System send popup message to actor, that register was succes

Exceptional:

	The data that input by user already in the database
	System will send pop up message to the actor that the account already taken

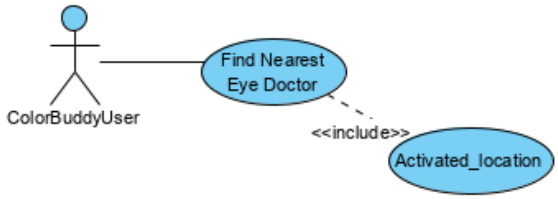
d. Use Case Diagram

D. USE CASE FIND NEAREST EYE DOCTOR

a. Name of Use Case: Find Nearest Eye Doctor
b. Actor: ColorBuddy User
c. Flow of Events

Actor click ‘find the nearest eye doctor’ menu	
	System will track user’s location and use google maps API to find the nearest eye doctor from actor location
	System will show the nearest eye doctor list

d. Use Case Diagram

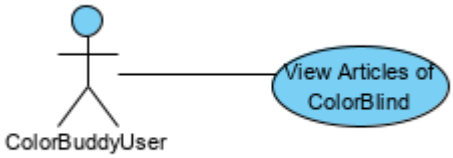


E. USE CASE VIEW ARTICLES OF COLOR BLIND

a. Name of Use Case: View Articles of Color Blind
b. Actor: ColorBuddy User
c. Flow of Events

Actor	System
Actor click 'article' page	
	System will execute actor's command, and display the article pages

d. Use Case Diagram

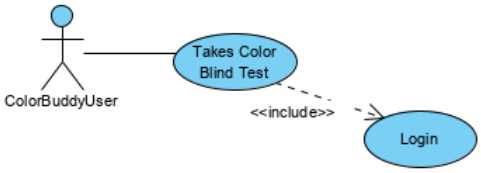


F. USE CASE TAKES COLOR BLIND TEST

a. Name of Use Case: Takes Color Blind Test
b. Actor: ColorBuddy User
c. Flow of Events

Actor	System
Actor click 'color blind test' page	
	System will generate actor command, and display the color blind test page
Actor fill out the answer	
	System will read actor's answer and generate the result of the test
	System will display the actor's color blind test result

d. Use Case Diagram

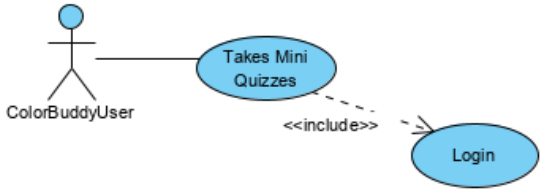


G. USE CASE TAKES MINI QUIZZES

a. Name of Use Case: Takes Mini Quizzes
b. Actor: ColorBuddy User
c. Flow of Events

Actor	User
Actor click 'mini quizzes' pages	
	System will generate actor command, and display the mini quizzes page
Actor fill out the answer of the quiz	
	System will read actor's answer and generate the result of the quiz
	System will display the quiz result

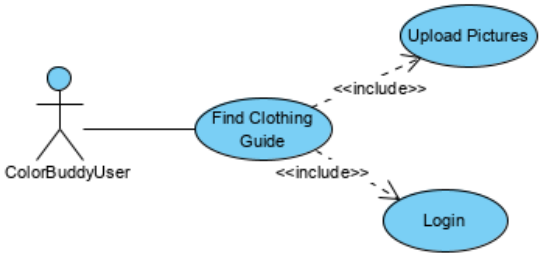
d. Use Case Diagram



H. USE CASE FIND CLOTHING GUIDE
   a. Name of Use Case: Find Clothing Guide
   b. Actor: ColorBuddy User
   c. Flow of Events

Actor	System
Actor click 'find clothing guide' pages	
	System will display 'find clothing guide' page
Actor input the image of the clothes that he/she want to match	
Actor click 'match the cloths'	
	System will generate actor commands and display the result

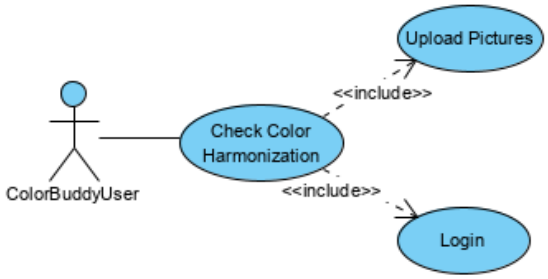
   d. Use Case Diagram



I. USE CASE CHECK THE COLOR HARMONIZATION
   a. Name of Use Case: Check the Color Harmonization
   b. Actor: ColorBuddy User
   c. Flow of Events

Actor	System
Actor click 'check the color harmonization' page	
	System will display the page
Actor pick the color that he/she want to harmonization	
	System will generate the color, and find the harmonization color
	System will display the result

   d. Use Case Diagram

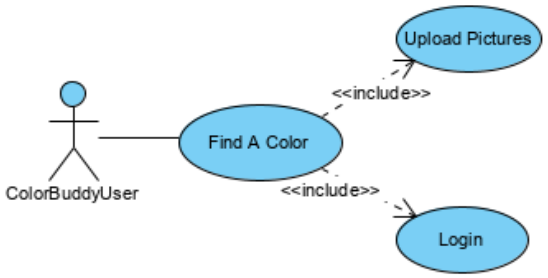


J. USE CASE FIND A COLOR
   a. Name of Use Case: Find A Color
   b. Actor: ColorBuddy User
   c. Flow of Events

Actor	System
Actor click 'find a color' page	
	System will display the page
Actor input the image and also the name of color that he/she wants to find	

	System will generate the input and then execute it to find the color
	System will display the result (the color that actor's want to find)

d. Use Case Diagram

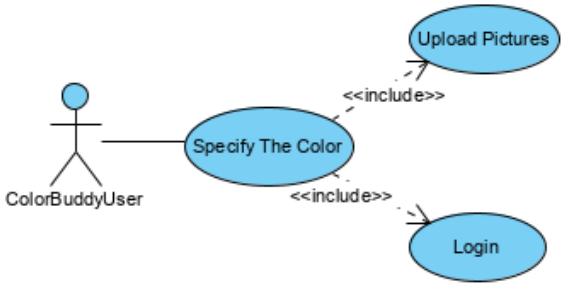


K. USE CASE SPECIFY THE COLOR

a. Name of Use Case: Specify the Color
b. Actor: ColorBuddy User
c. Flow of Events

Actor	System
Actor click 'Specify the Color' page	
	System will display the page
Actor input the image that he/she wants to specify	
	System will read the image and execute the color that occur on the image
	System will display the result

d. Use Case Diagram



REFERENCES

[1] https://www.lexico.com/en/definition/colour

[2] http://www.arcsoft.com/topics/photostudio-darkroom/what-is-colorspace.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

[7] https://uxplanet.org/algorithm-for-automatic-harmonious-color-selection-for-the-image-fc26dde69ca1

[8] http://www.websiteoptimization.com/speed/tweak/color-harmony/

[9]

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.