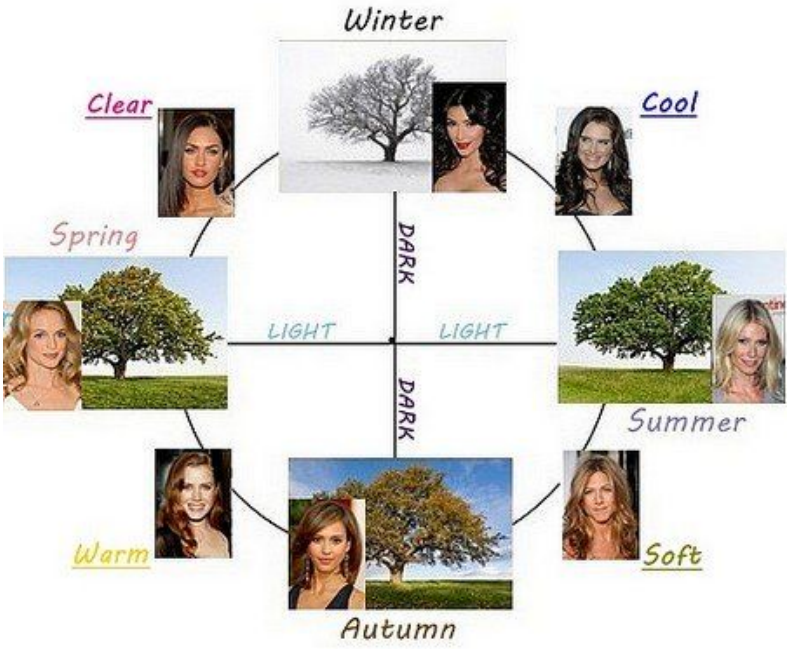
GitHub Username: sosegon

Seasonify

Description

In the fashion and cosmetic industries, the seasonal color of a person is used to determine the colors of clothing and makeup that best match a person's skin complexion, eye color, and hair color. This approach categorizes people into four categories:

- Winter
- Spring
- Summer
- Autumn



Source: http://www.thechicfashionista.com

Unfortunately, this information is not known by most people. They do not about seasonal colors, even worse, they do not how to determine theirs. This lack of information leads people to wear clothes that do not suit their characteristics.

Seasonify provides a way to easily determine the seasonal color of people by analyzing their faces and, based on the result, provide color information for clothes that best suits their characteristics.

Intended User

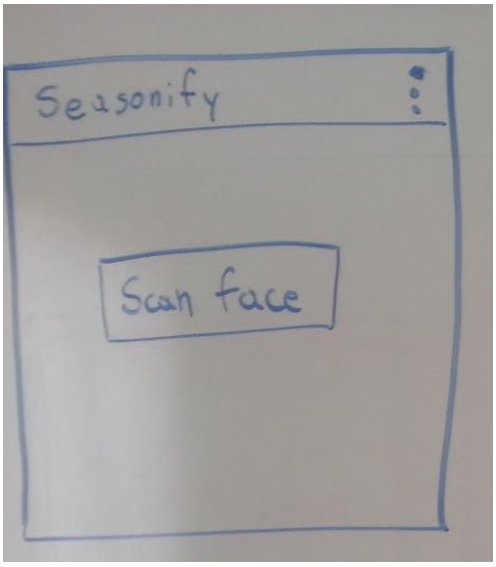
The application is intended to be used by people interested in wearing clothes that suit their personal characteristics.

Features

- Starts the camera to detect a face to be analyzed.
- Predicts the seasonal color of a person.
- Provides color information for clothes that better suits for a person.

User Interface Mocks

Start Activity

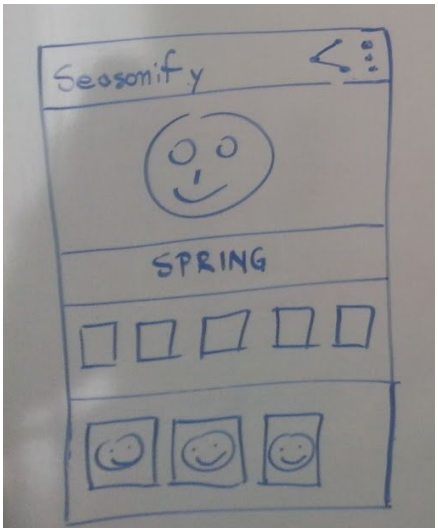


This is the first activity of the application. It appears the first time the application is launched. It contains a button that launches the camera activity when tapped. It also contains an app bar with the logo of the application and options menu.

Camera Activity

In this activity, the front camera is activated it covers the entire screen of the device.The user has to bring the camera close her face and tap the scan button. The application will take several photos that will be used to determine the seasonal color of the person.

Results Activity

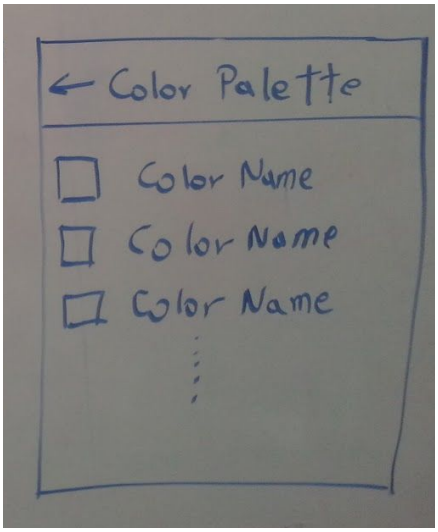


Once the application has processed the images, the results are displayed in a new activity. This one depicts one of the photos taken by the application along with the corresponding seasonal color.

It also shows a palette with the colors for clothes and makeup that better fit the characteristics of the person. Finally, it presents examples of famous people with the same personal color.

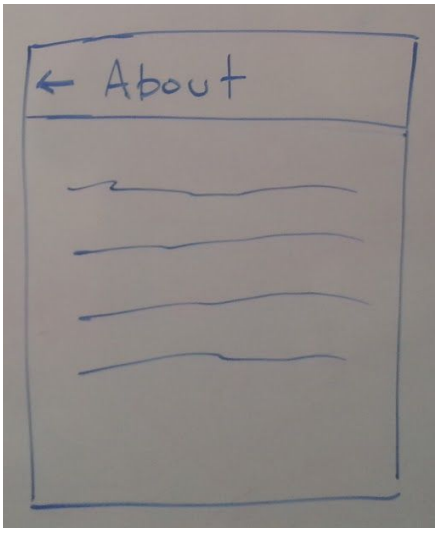
The application bar contains an item to share the results, and menu options with two items: About, and Scan Face. About launches an activity with information of the app. Scan Face starts the camera activity.

Palette Activity



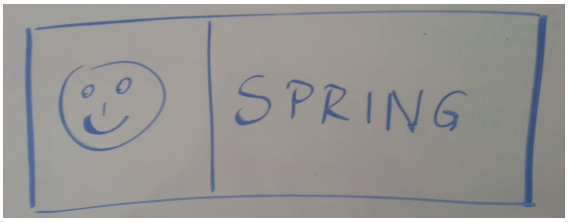
Tapping in the color palette shown in the previous activity launches a new one with a more detailed palette that includes the name of the colors.

About Activity



The application also contains an about section with information related to the theory behind the process and the application itself.

Widget



The widget depicts the image of the last face analyzed and the corresponding prediction.

Key Considerations

How will your app handle data persistence?

The application will have a content provider to store the information displayed in the Results Activity:

- Location of the images taken to predict the seasonal color.
- Colors displayed in the palette.
- Location of the pictures of other people with the same predicted seasonal color.

Describe any corner cases in the UX.

The application is designed so the user has to do as fewer actions as possible. For instance, the camera activity will display text to guide the user in order the take good images. In this activity, the application will detect a face and will depict a circle around it, this way the user can be confident enough to scan the face.

The navigation is done with the back and up button (About and Color Palette activities).

Describe any libraries you'll be using and share your reasoning for including them.

I will use the following libraries:

- OpenCV: This library provides the tools for image processing. It has capabilities to detect the face in the camera activity. Also, this library has methods to process the images so they have the proper format to be used in the classifier.
- Android TensorFlow support: This component has the tools to use the model to make the predictions.
- Picasso: This library provides capabilities to display images in the results activity.
- Schematic: This library simplifies the creation of a content provider with the use of annotations

Describe how you will implement Google Play Services.

I will use Firebase for the following purposes:

- Remote storage: To save the assets used in the application including the model to predict the colors.
- Notifications: To engage the users in utilizing the application.
- Analytics: To determine how users are using the application and what kind of improvements can be done

Next Steps: Required Tasks

Task 0: Model creation

This step is not related to Android Development, however, it is necessary for the main feature of the application: predict the seasonal color of a person. This step is all about the creation of a Machine Learning model that will be used to make the predictions.

Task 1: Project Setup

- Create the git repository for the project.

- Set the corresponding libraries that are the core of the application.
- Set a firebase project and attach it to the application.

Task 2: Implement UI for Each Activity and Fragment

- Create the layouts for different devices and orientations: phones and tablets.
- Mockup activities and fragments.
- Wire up activities and fragments.

Task 3: Design content provider

- Create the java classes for the elements to be handled by the content provider.
- Create the content provider.

Task 4: Implement core features

- Create the logic to detect a face with the camera.
- Create the logic to preprocess an image to pass it to the classifier.
- Pass the results and depict them in the corresponding activity.

Task 5: Fine Tune application

- Implement concepts of Material Design.
- Design appealing logo for the application.