GitHub Username: Sandeshkaranam

Quotes for photos

Description

Do know to describe this image which you want to share on social network? Here the answer:
Quotes for photos is an application to suggest the best quotes to describe the photo based on image analysis.
An image will be analysed using Microsoft cognitive analysis API and allows users to select the tags. Based on the selected tags, it will suggest the list of quotes using fraze.it API and user can select one of them. Finally, compose the message based on selected quote and image to share on social media like Facebook, twitter and others.

Intended User

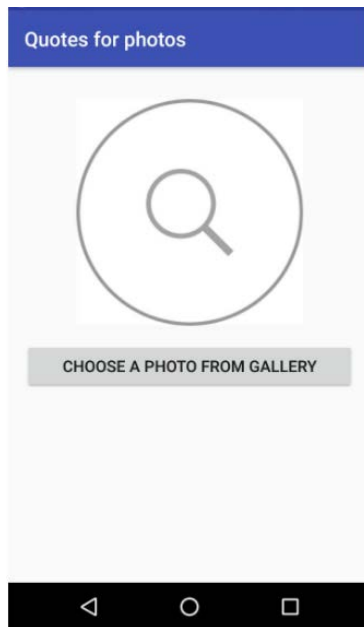Blog writes, Facebook, twitter and other social networking users

Features

List the main features of Quotes for photos:
- Suggests the quotes based on photos
- Allows users to customization search of quotes

- Allows to edit the text/description of the image.
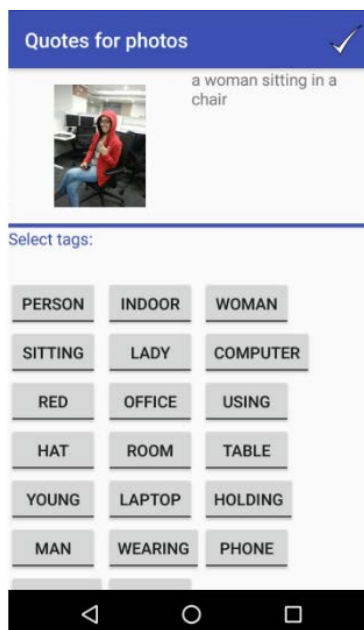- Share the image and text/description on social networking apps

User Interface Mocks

Screen 1
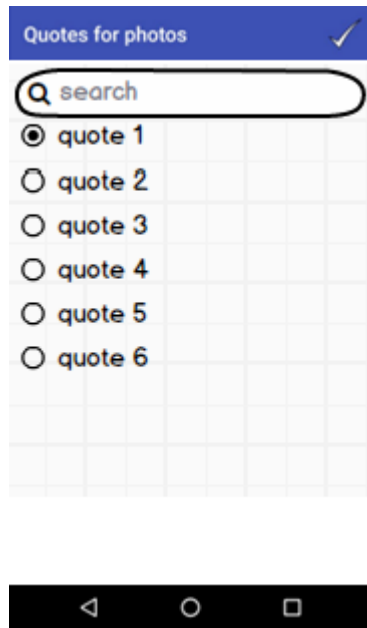

Choose an image from gallery and it will Uploaded for an image analysis through Microsoft cognitive image analysis API

Screen 2

It will provide basic description and also tags, allows users to select the tags which are interested in.

Screen 3

It will prompt the quotes based on selected tags and allows user to search manually. They can select the quote from the list.

Screen 4

Next activity will contain image preview along with selected quotes. It will allow the user to edit the description and finish the composition of the message. They can share the image and composed message on social media like Facebook, Instagram and others.

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

Will connect to the Microsoft vision and fraze.it API to get data but no need to persist the data because app is dynamic based on the user inputs. Use a content provider to load the data for widget which display quote of the day.
App will use async tasks to connect to the api and get the data.

Describe any corner cases in the UX.

If user opens app when no network connectivity, then proper error messages are shown
Try again or cancel button will be prompted.

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

`Microsoft-projectoxford:vision – compress the image before sending to api and store the JSON responses.`

Describe how you will implement Google Play Services.

Admob to display the advertisements in the application
Google+ for the sharing the content.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

Setup all required third party libraries and APIs required for the project.
Ex: Configure dependencies for Microsoft-projectoxford, google play services etc..

Get the API_KEY for the project.
Setup any development environment required for the project, setup all the emulators etc.

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:
- Build UI for fetch the image from gallery
- Build UI for display fetched image, description and tags based on response from the API
- Build UI to list the quotes and search bar
- Build UI for finish the composition of the message along with image.

Task 3: Handle API call response Error and network Error

Verify the bad response and network connectivity lost errors, prompt correct error message.

Task 4: Custom search of the quotes
Allowing users to search the quotes and display the results. Radio button select only one quote from the list

Task 5: Share on social network
Implement share intent which contains the image and description of it.