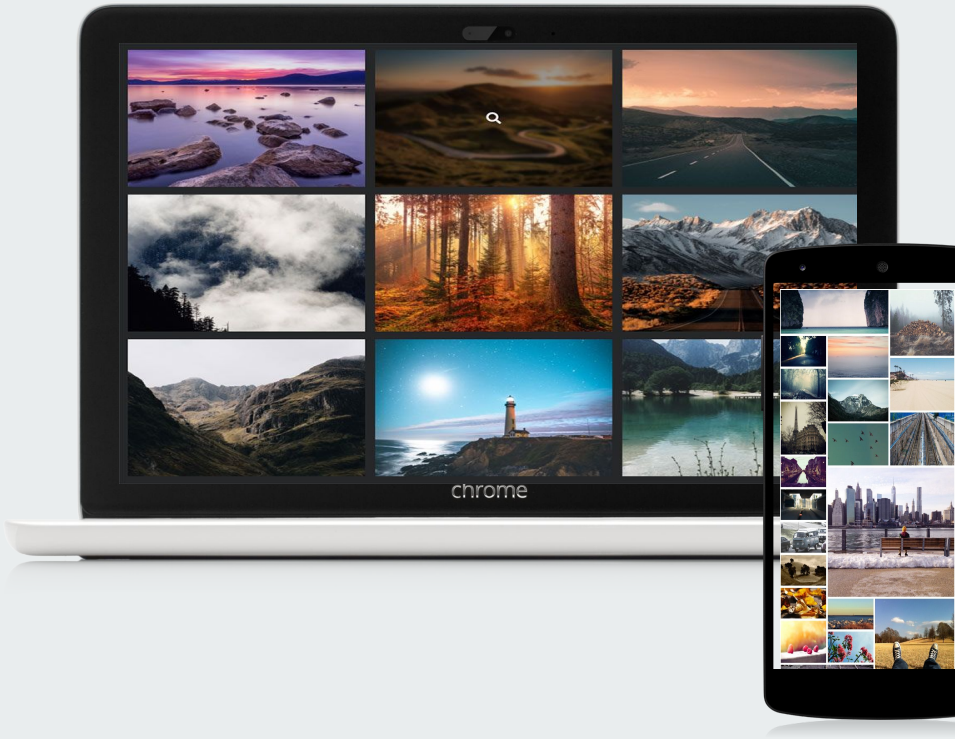
SmartGallery IITISOC

Presented by-MISC1

TEAM MEMBERS

- JYOTISHNA BAISHYA
- ANANYA SAHU
- ABHISHEK DHANGAR
- ADHIP S. DEY



PROJECT DESCRIPTION

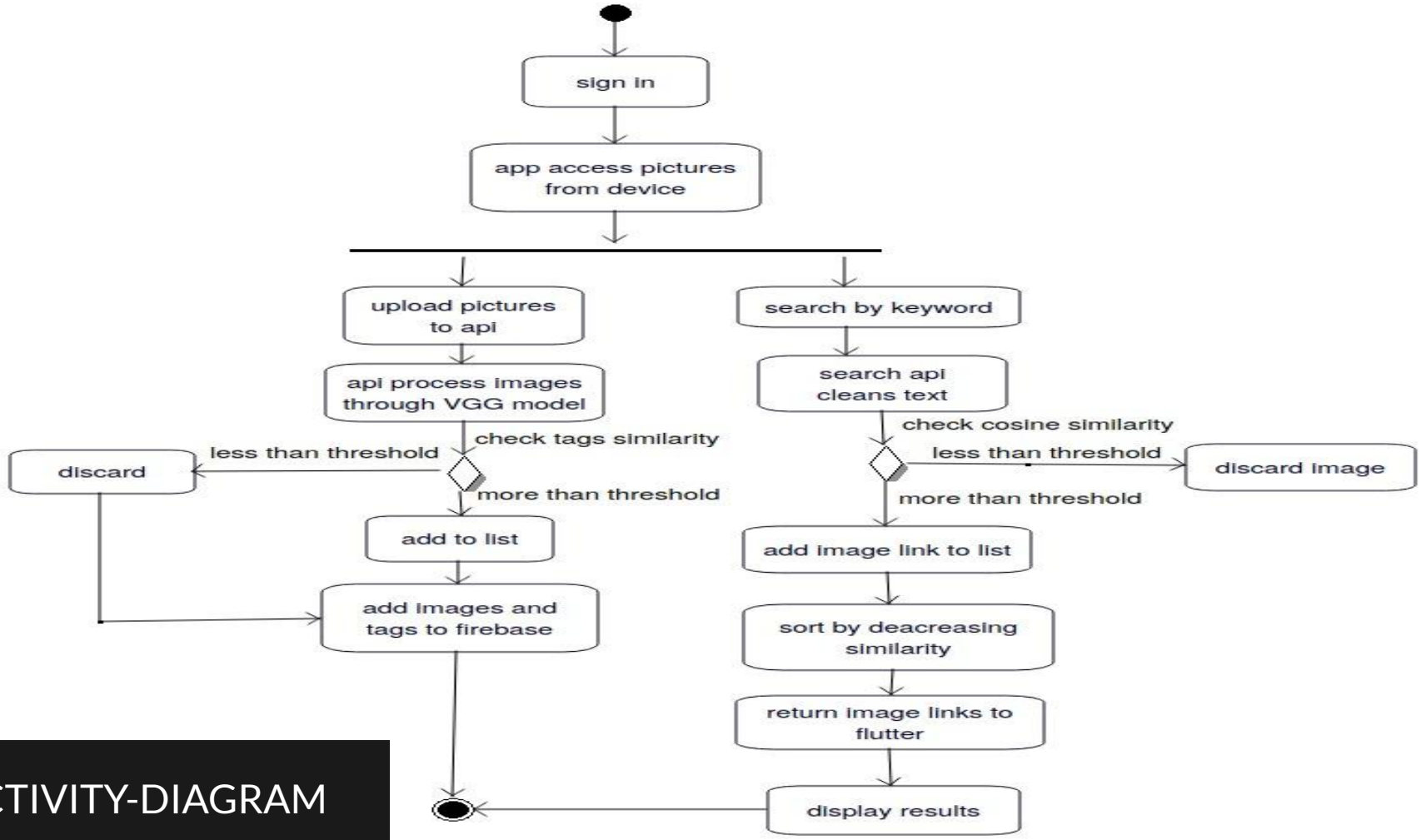
A photo gallery app which organises your photos, helps you find them in a easier way by allowing you to search photos using words in a way such that picture with maximum similarity with searched text appears on top of image grid.

TECH STACK

- **Back-end- Flask**
- **Front-end- Flutter**
- **Database- Firebase**

Tasks

- Generation of image tags.
- Preprocessing of text.
- Comparing text and tags vector.
- Upload images to firebase storage
- Storing tags and image link in firestore.

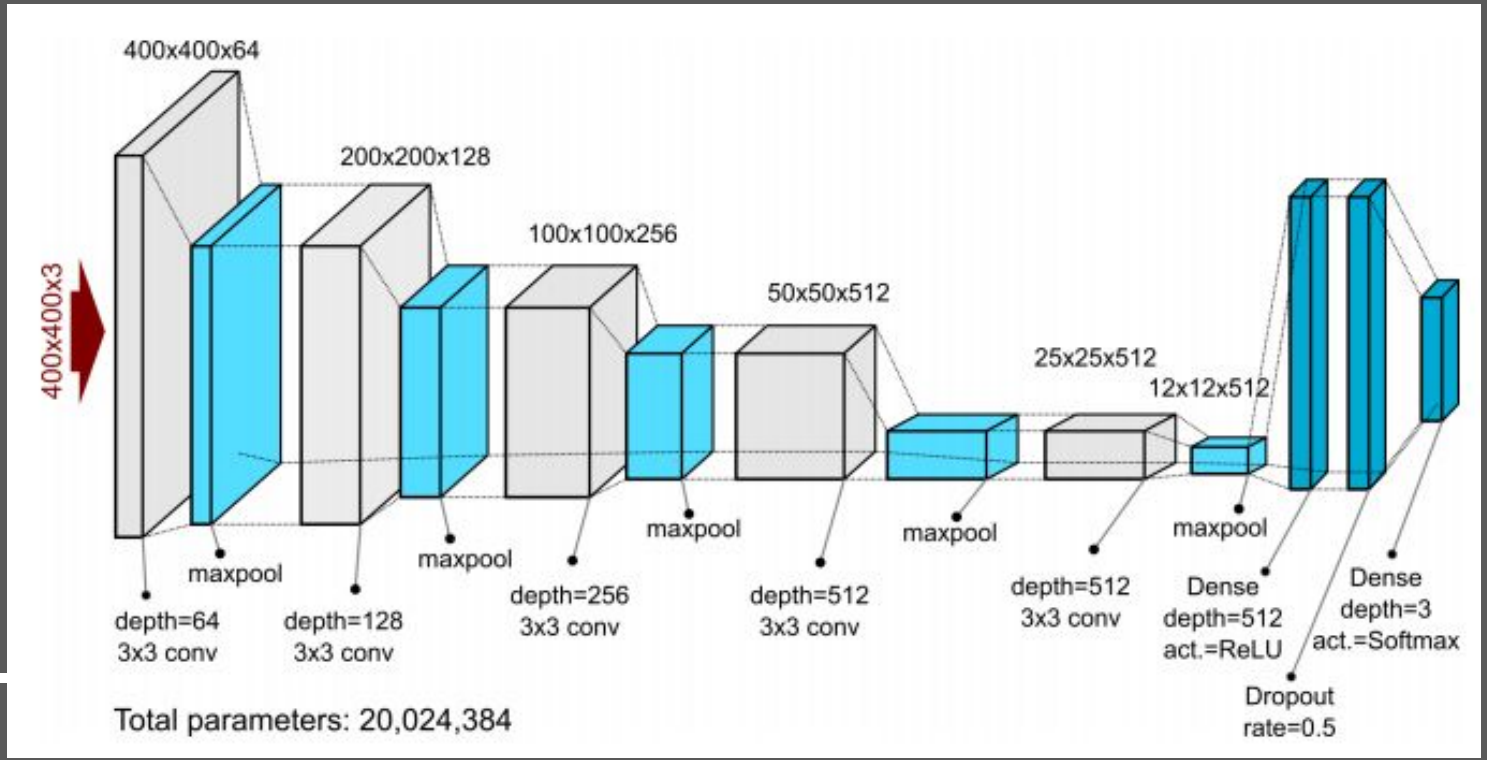


ACTIVITY-DIAGRAM

Image Tags generation

- Image classification is a process of putting tags or labels on image.
- Used pretrained VGG19 image model to classify images.
- VGG 19 is a convolutional neural network that is 19 layers deep. We have used a pre trained version of the network trained on more than million images from large Image Net dataset.
- The pretrained network can classify images into 1000 object categories.

STRUCTURE OF VGG19



CLEANING TEXT

Performed following operations with NLTK on searched text-

- Tokenize - splits string of text to array of words.
- Lemmetize - convert words to its base form . For example- bags to bag, eating to eat ,etc.
- Remove stopwords- stopwords like a, an, the, is, am, are, would, should, etc are removed.

COMPARING TAGS AND TEXT VECTOR

- Used **cosine similarity** algorithm for making comparison.
- Use word2vec model to generate word embeddings.
- Word2vec creates vector that are distributed numerical representations of word features such as the context of individual words.
- We compare the cleaned searched text with all the image tags vector and calculate cosine similarity for all of them.
- The tags vector with cosine similarity greater than fixed threshold will be displayed in image grid.
- Also they are displayed in descending order of their similarity values.

Database-Firebase

- We have used Firebase as our database it is Google-backed application development software that enables developers to develop iOS, Android and Web apps.

- The platform provides developers an API that allows app data to be synchronized across clients and stored on Firebase's cloud. In our case, firebase is connected to both flutter frontend as well as to python server.

- It's cloud storage service is useful for storing user images.
- Using firebase enabled users to connect to our app directly through their google account.

Using Firebase Storage

Images are accessed by app as well as server. Hence, they are stored using cloud storage.

- Cloud storage allows access to images from both server-side and client-side. Hence, data exchange becomes efficient as payload size decreases significantly using image links.
- Allows images to be associated with their rightful users in a secure manner.
- Firestore database stores links to these images along with their tags.

The image links from google storage api allows images to be displayed in image grid directly from google storage bucket in flutter.

Authenticating Users Through Firebase

For User registration we are using Authentication feature of firebase which allows an app to securely save user data in the cloud and provide the same personalized experience across all of the user's devices.

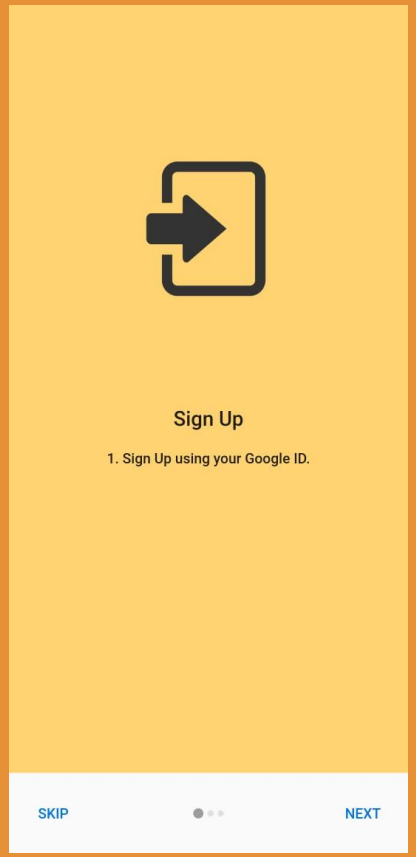
Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to our App.

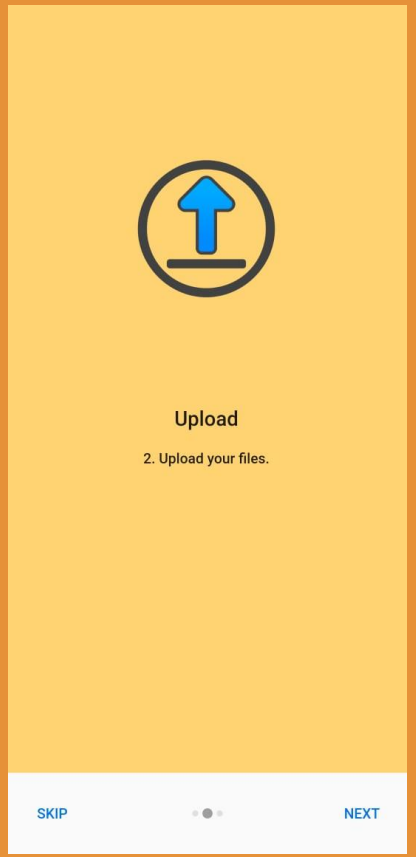
From various options available we are using Google sign-in option to authenticate our users.

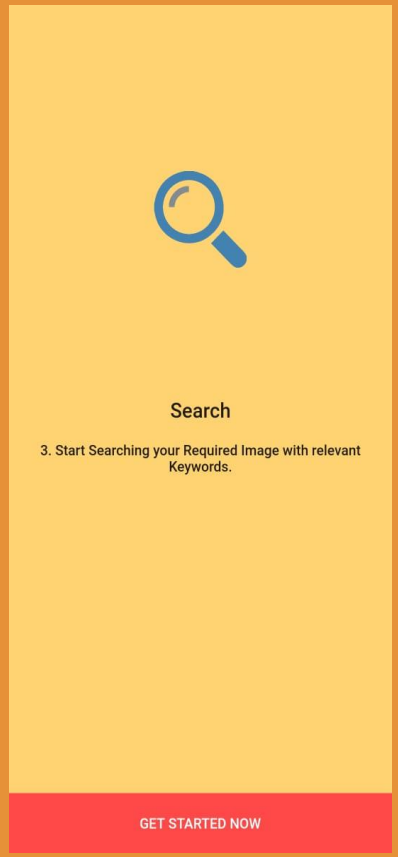
FrontEnd

The User Interface is created using Flutter.

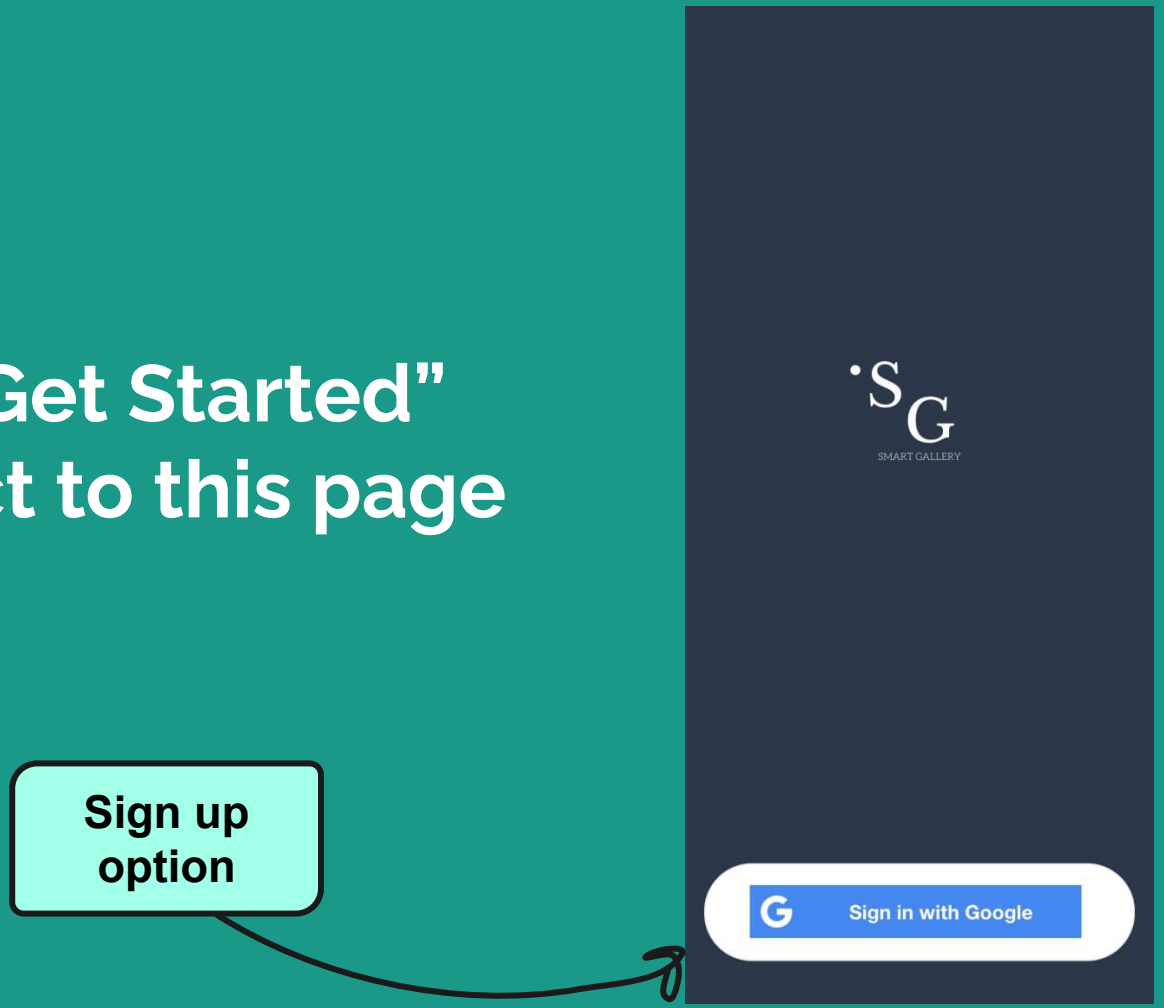
Intro-slider pages



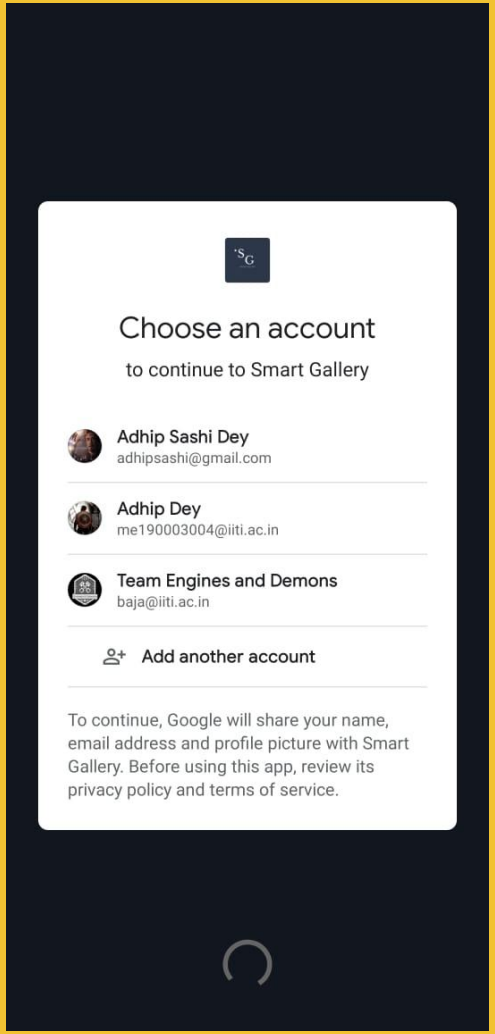




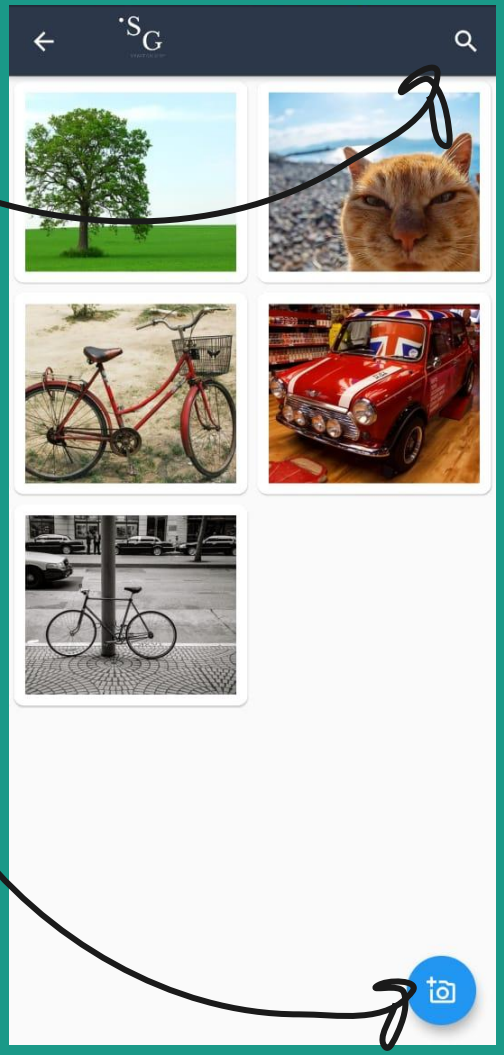
On clicking “Get Started” app will direct to this page



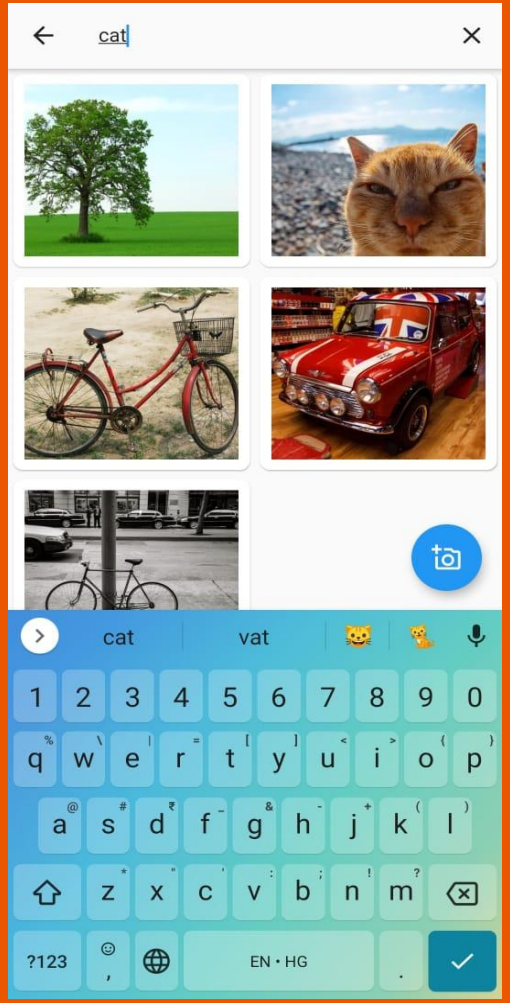
When user Click on sign in button app gives option to sign in with google account

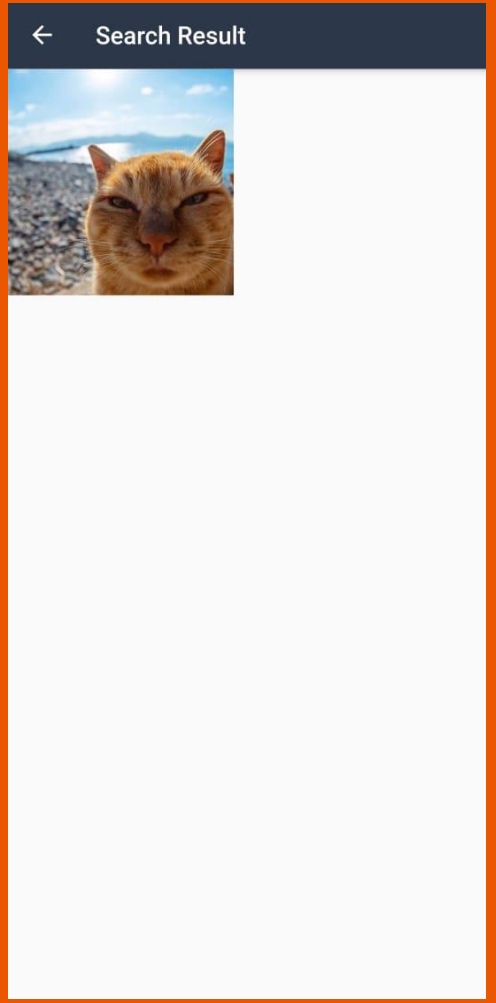


Home Screen
This is the Home Page



Search result will appear on this page






thank you
Ins
NumLk
Del
ScrLk
Shift
PgUp
PgDn
End
Home