My Photos Search

Project Coordinator: Ron Krasik

rkrasik2@illinois.edu

Team #75 (team of 1)

Overview of **My Photos Search** Project

- Problem definition
- Novelty of my project
- Google Vision API
- General approach
- Implementation Details (+ Architecture)
- System snapshots
- Installation options
- Key learnings and further extension

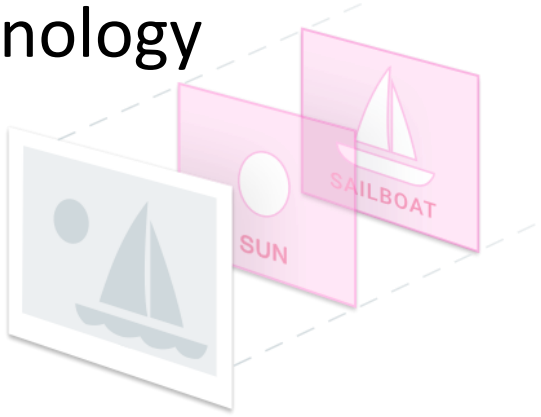
Problem definition

- As of today, end-user has quite limited ability to perform **content search** for images dataset that is stored on his/her computer or on one of commonly known service providers (Google Drive, Dropbox, Instagram, Facebook, …)
  - *Disclaimer: basic attributes search is typically available (picture date, picture name, picture size, …)*

Novelty of my project

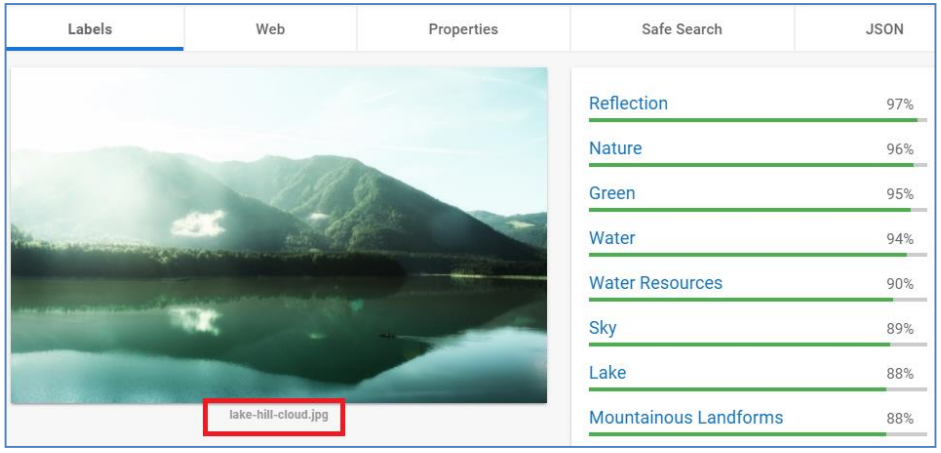
- The only such tool existing today is usual file search provided by Windows and it is quite limited as it allows only basic search based on “physical” attributes of the picture (date, size, type, name…)

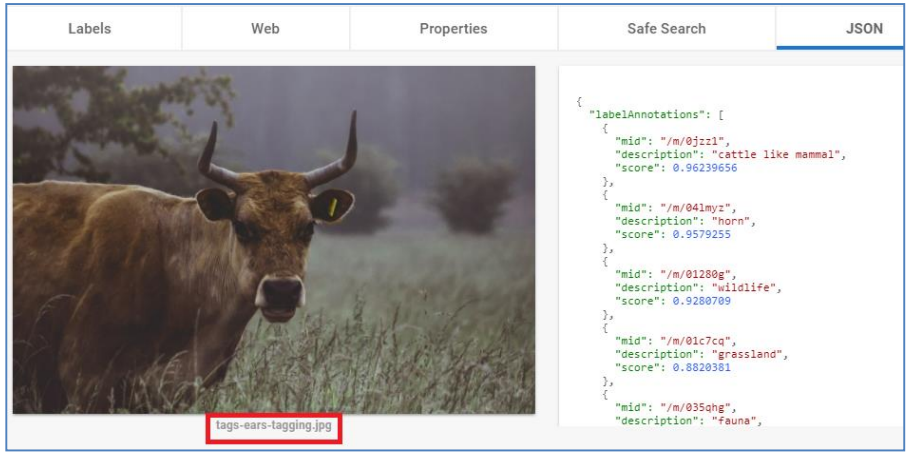
- **My Photos Search** improves the search immensely, as it introduces efficient content search, while implementing state-of-the-art Google Vision API technology



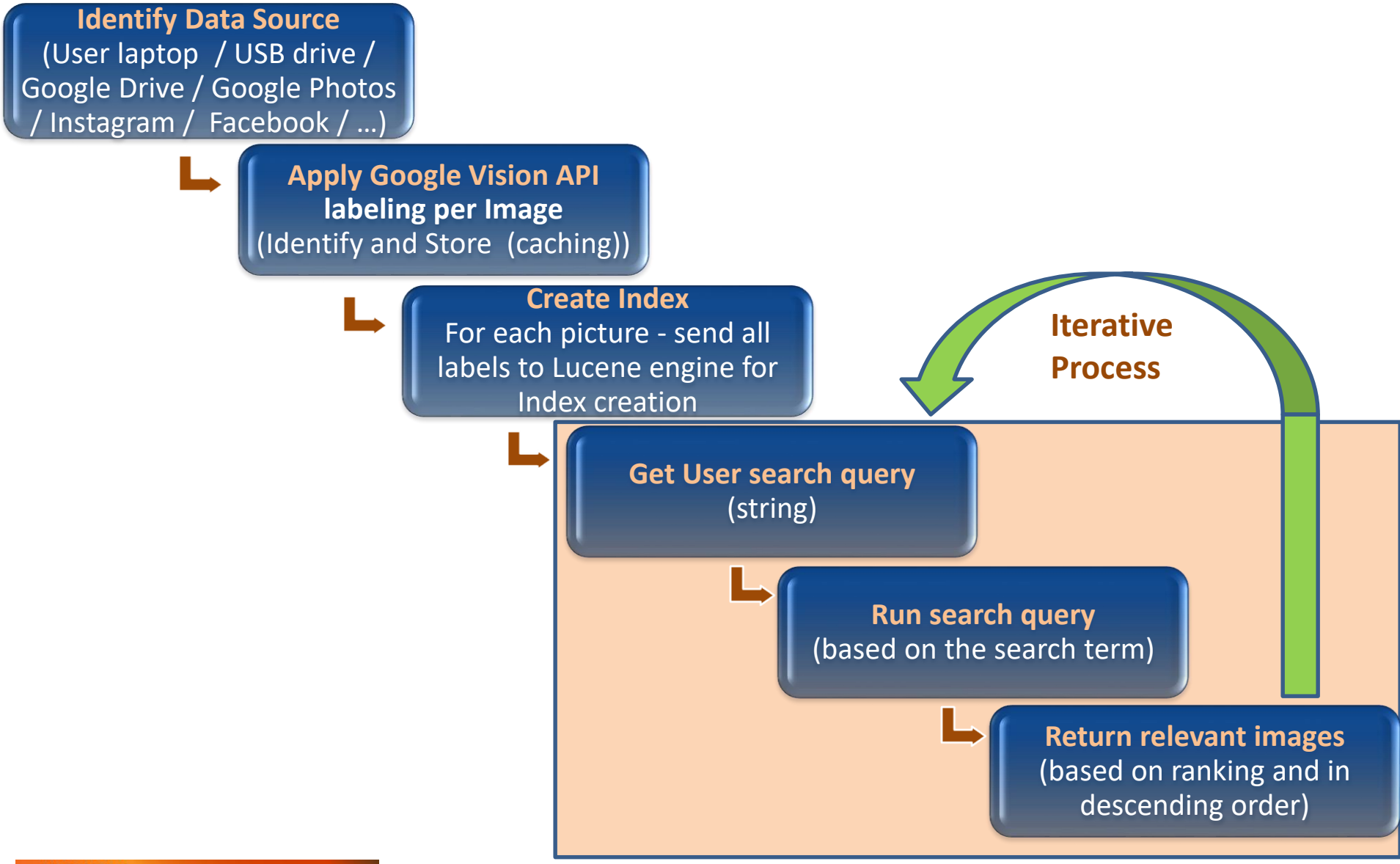
Google Vision API

- Enables to **understand the content of an image** by encapsulating **powerful machine learning models** in an easy to use REST API
- Rapidly **classifies** images into thousands of categories (e.g., "boat", "lion", "Eiffel Tower"), **detects** faces with associated emotions, and **recognizes** printed words in many languages. Hereunder are main general classifications/capabilities:
  - Label/Entity Detection picks out the dominant entity (e.g., a car, a cat) within an image
  - Optical Character Recognition to retrieve text from an image
  - Safe Search Detection to detect inappropriate content within your image
  - Facial Detection can detect when a face appears in photos, along with associated facial features
  - Landmark Detection to identify popular natural and manmade structures
  - Logo Detection to identify product logos within an image





General approach



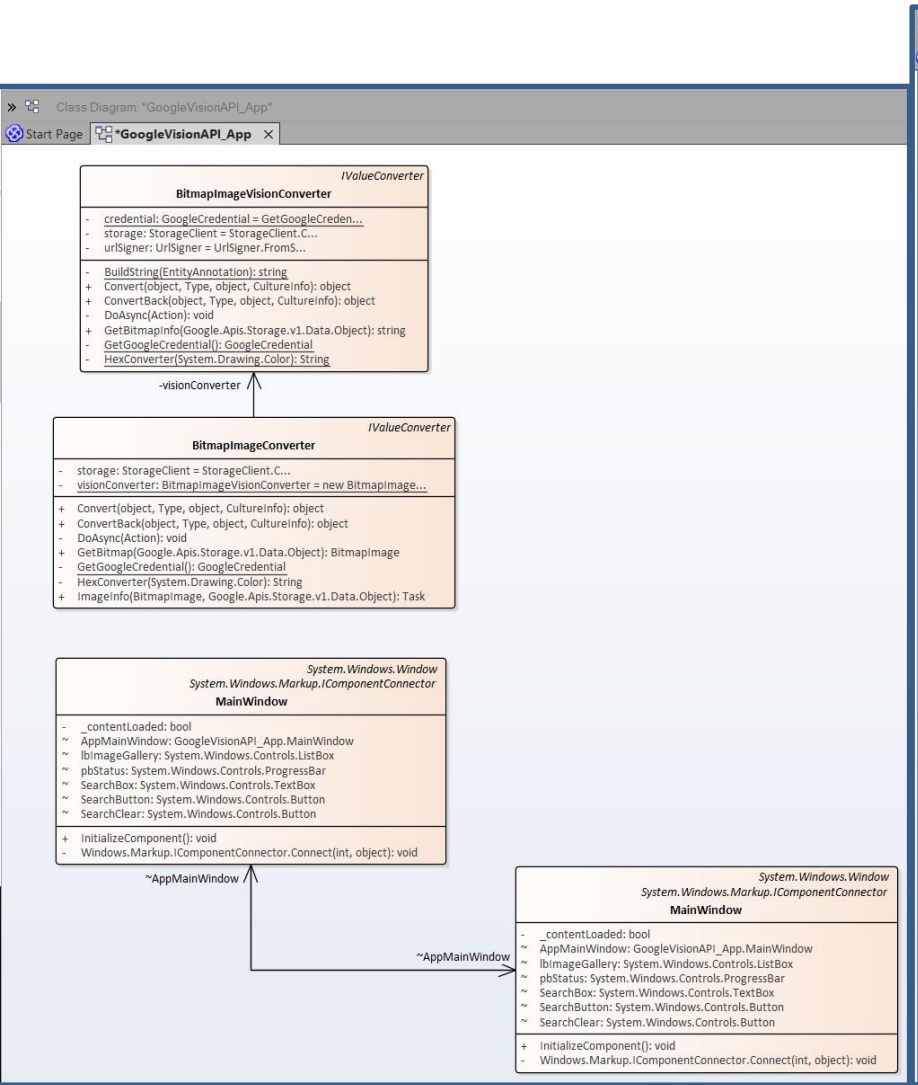
Implementation details

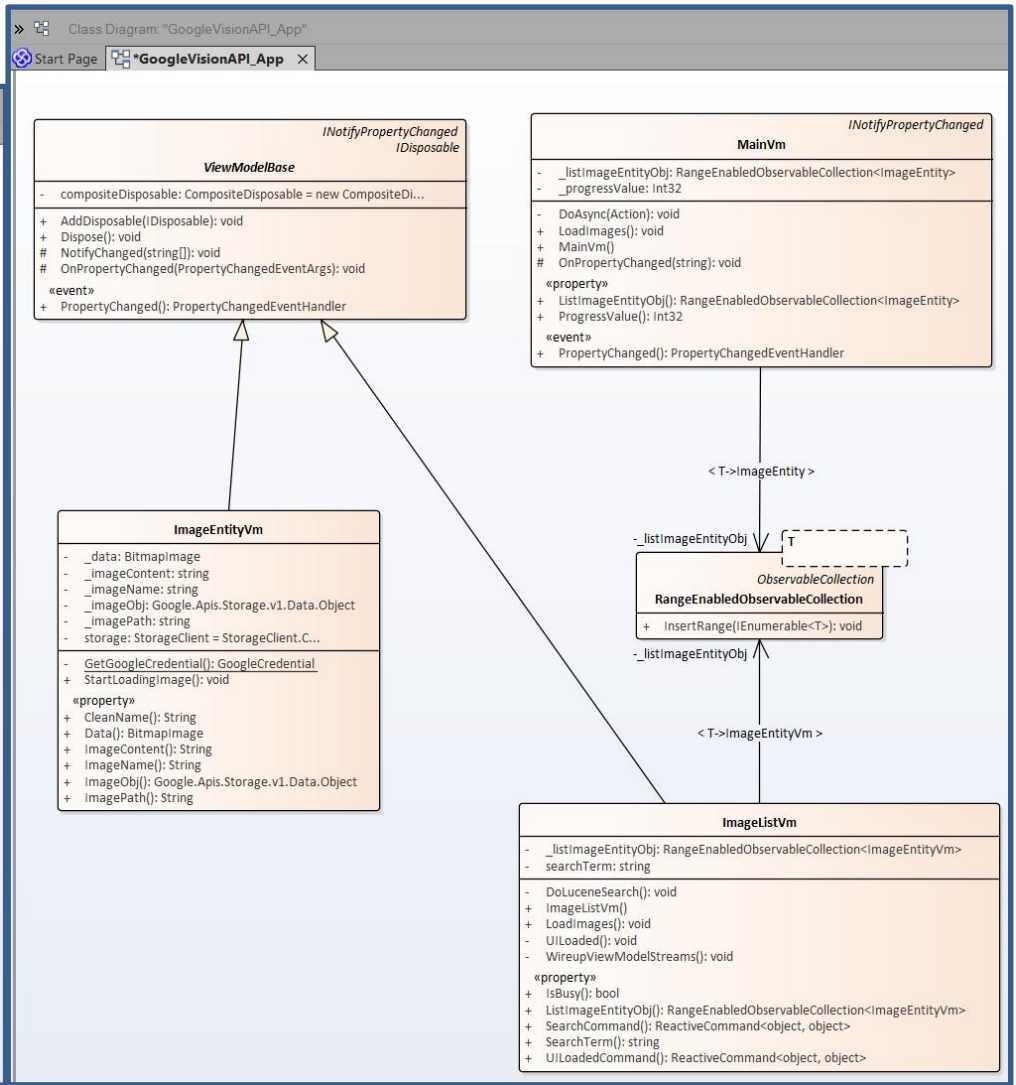
- Sample dataset was used by downloading ~6,300 pictures from http://visualhunt.com (free license)

- Google Cloud Storage was selected to host the images

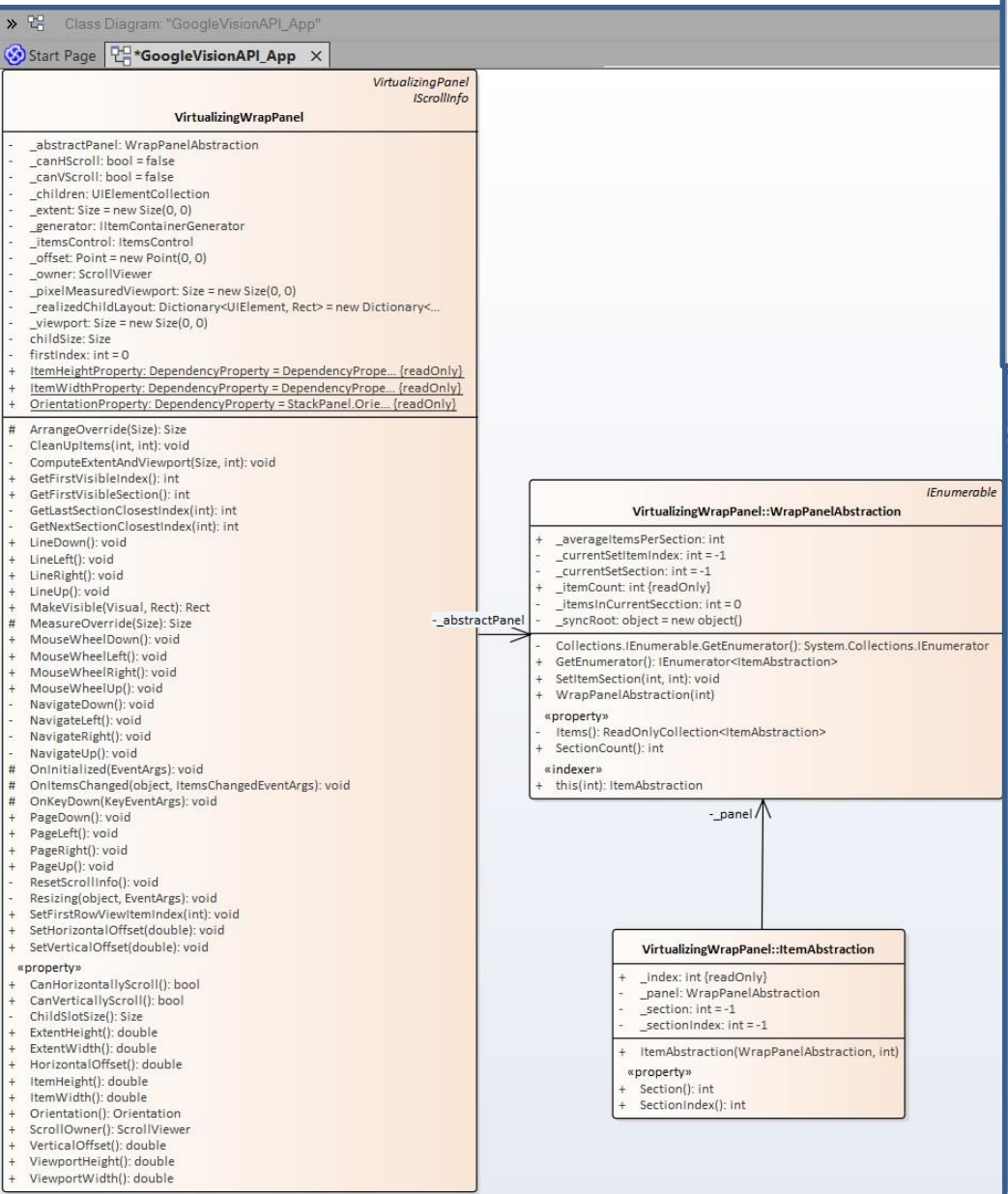
- Implementation - WPF (IDE = Visual Studio 2017)

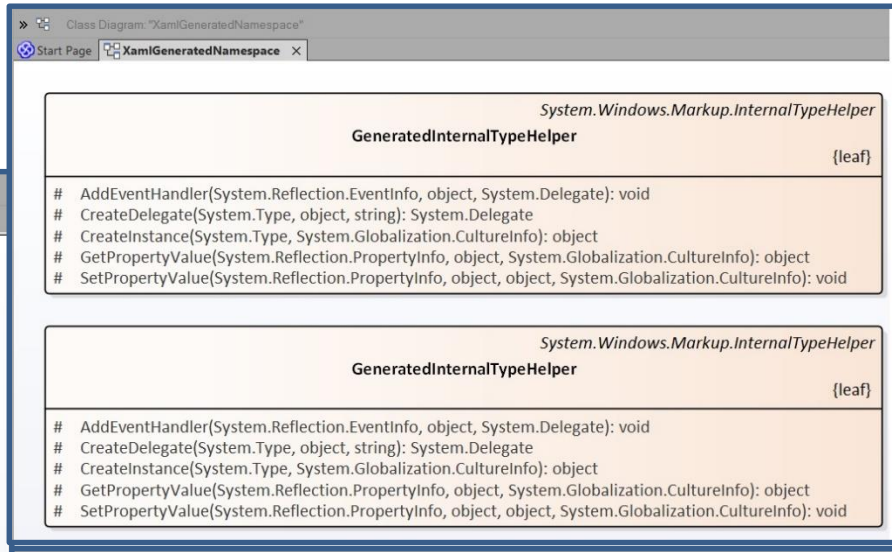
System Architecture (Class Diagram)

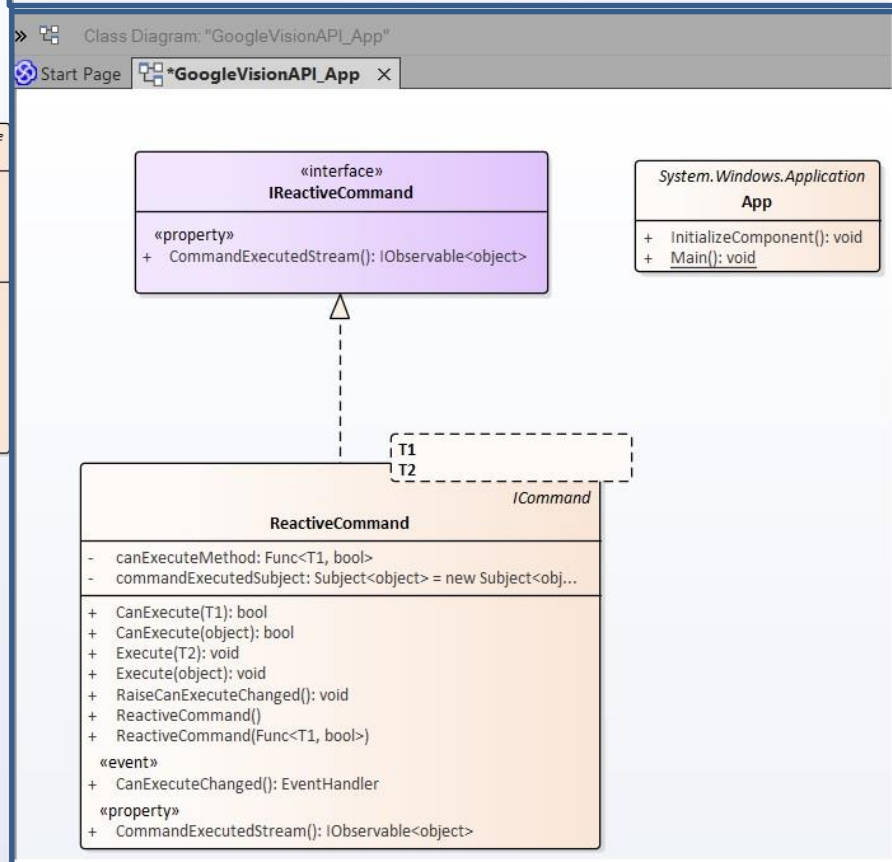




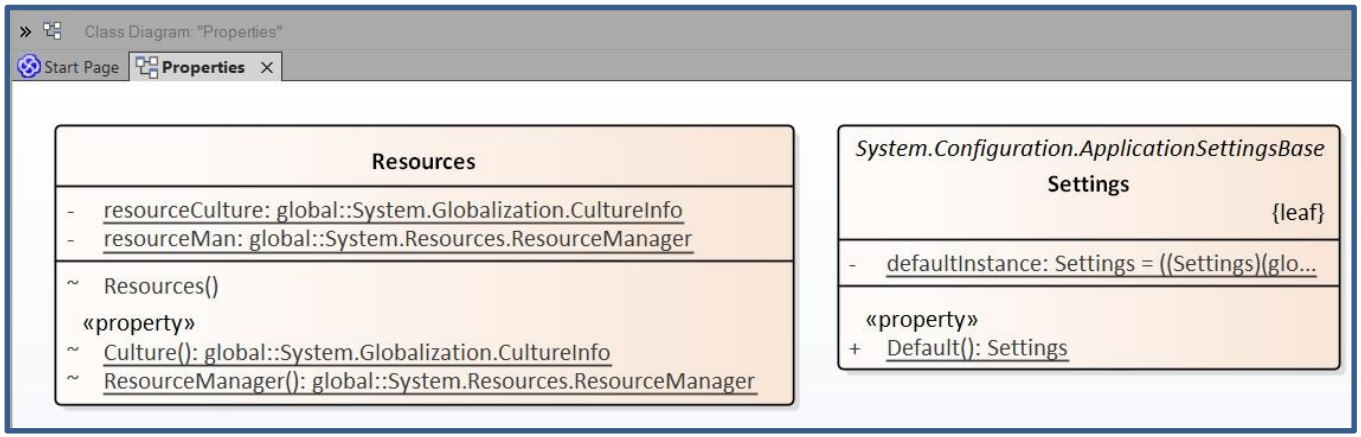
System Architecture (Class Diagram)

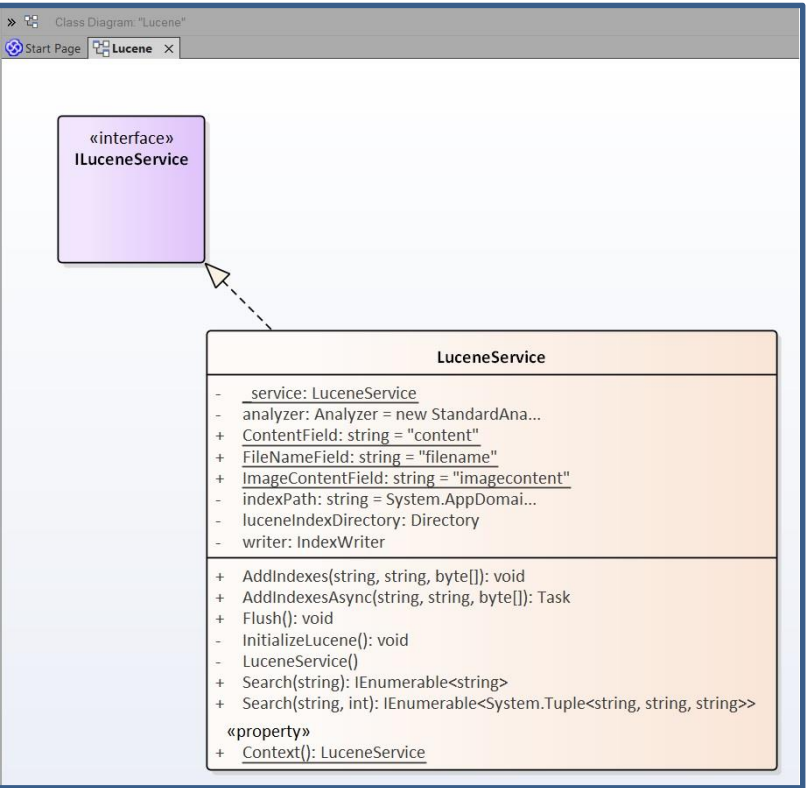


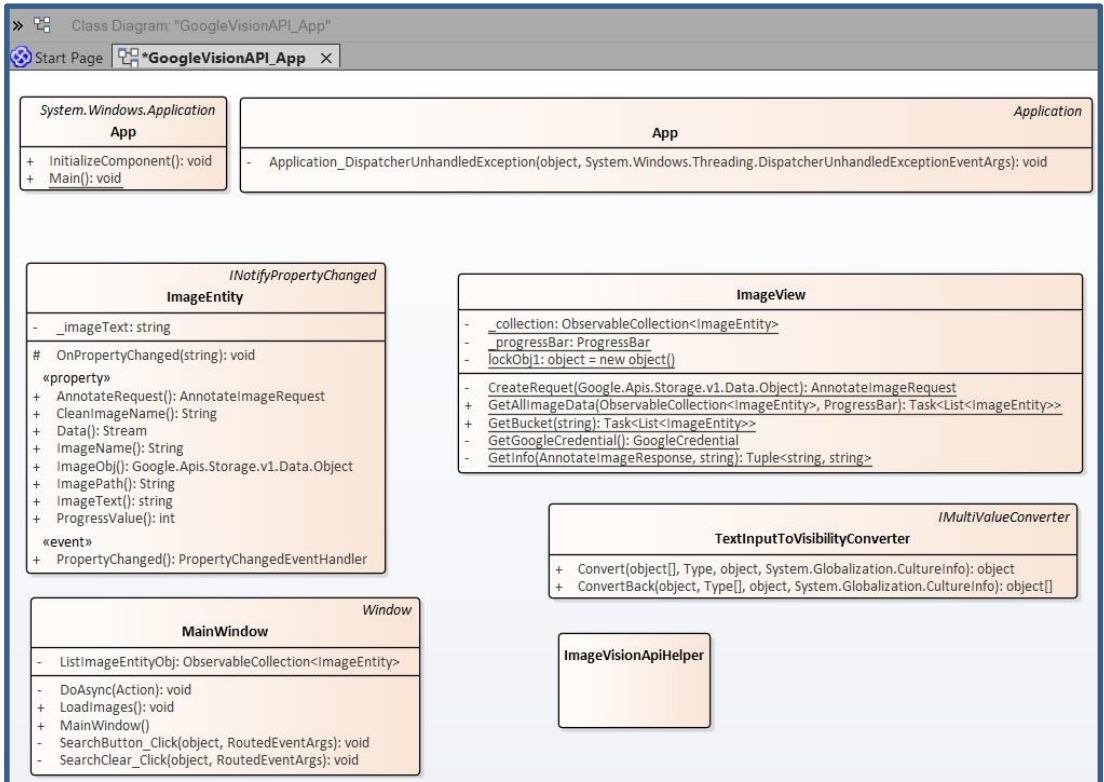




System Architecture (Class Diagram)



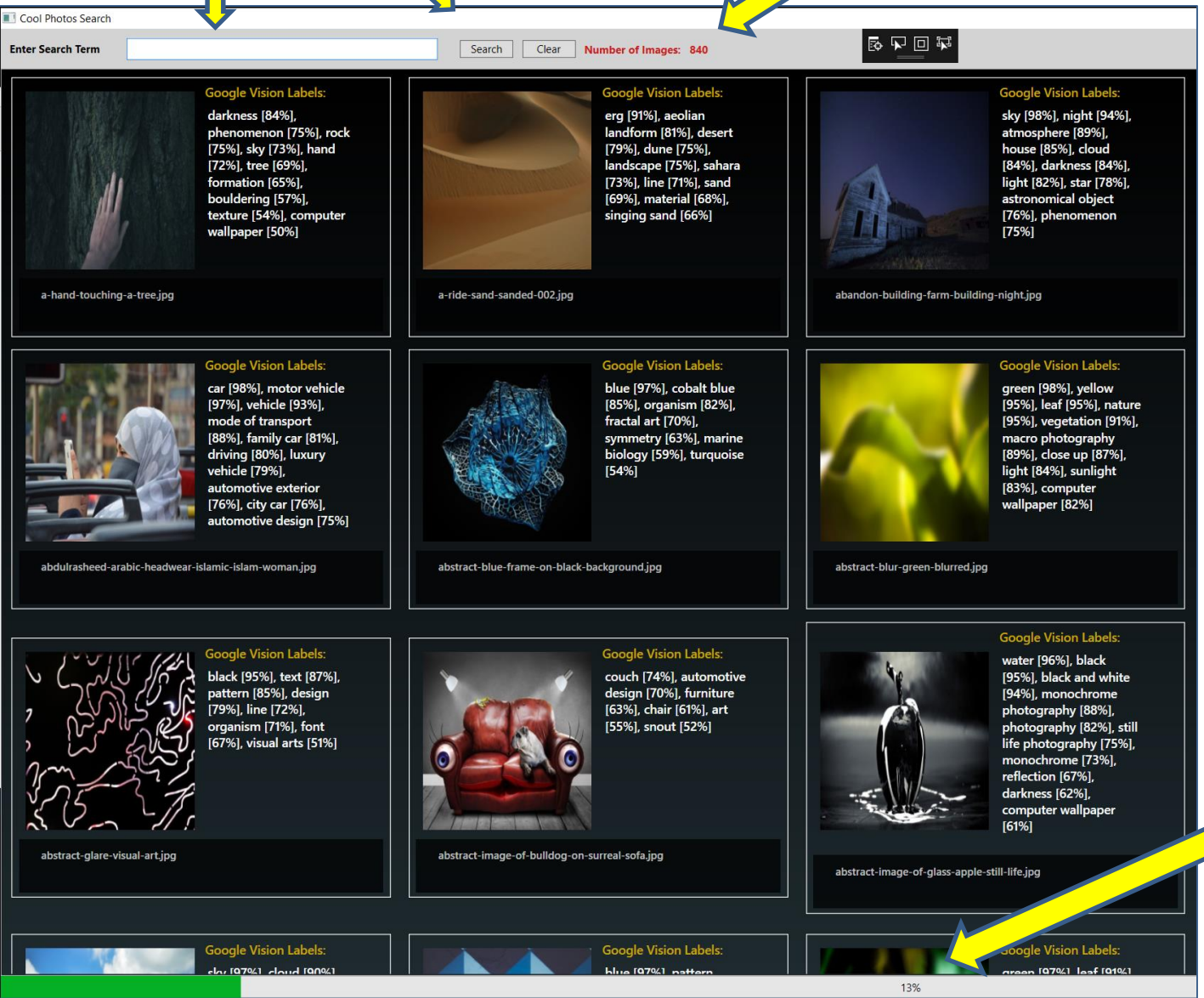




System snapshots – Main screen

Search will activate search based on the provided string query

Loading pictures from Google Cloud (default ordering is alphabetical, number of currently loaded pictures is shown on the top bar - 840 in the example)



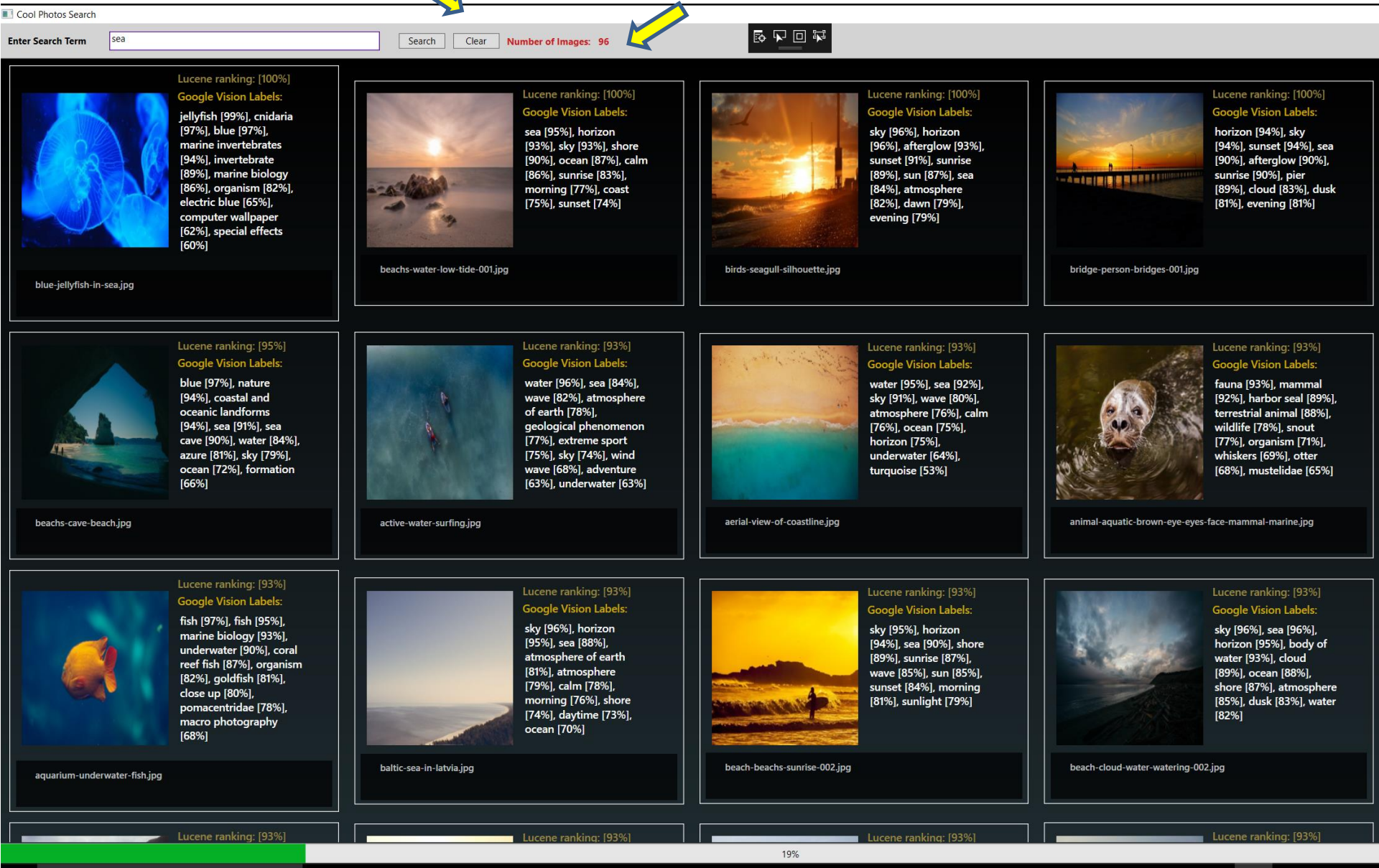
Each picture shows relevant Google Vision labeling

Global load progress bar is shown at the bottom of the page (13% in the example)

System snapshots – Search (sea)

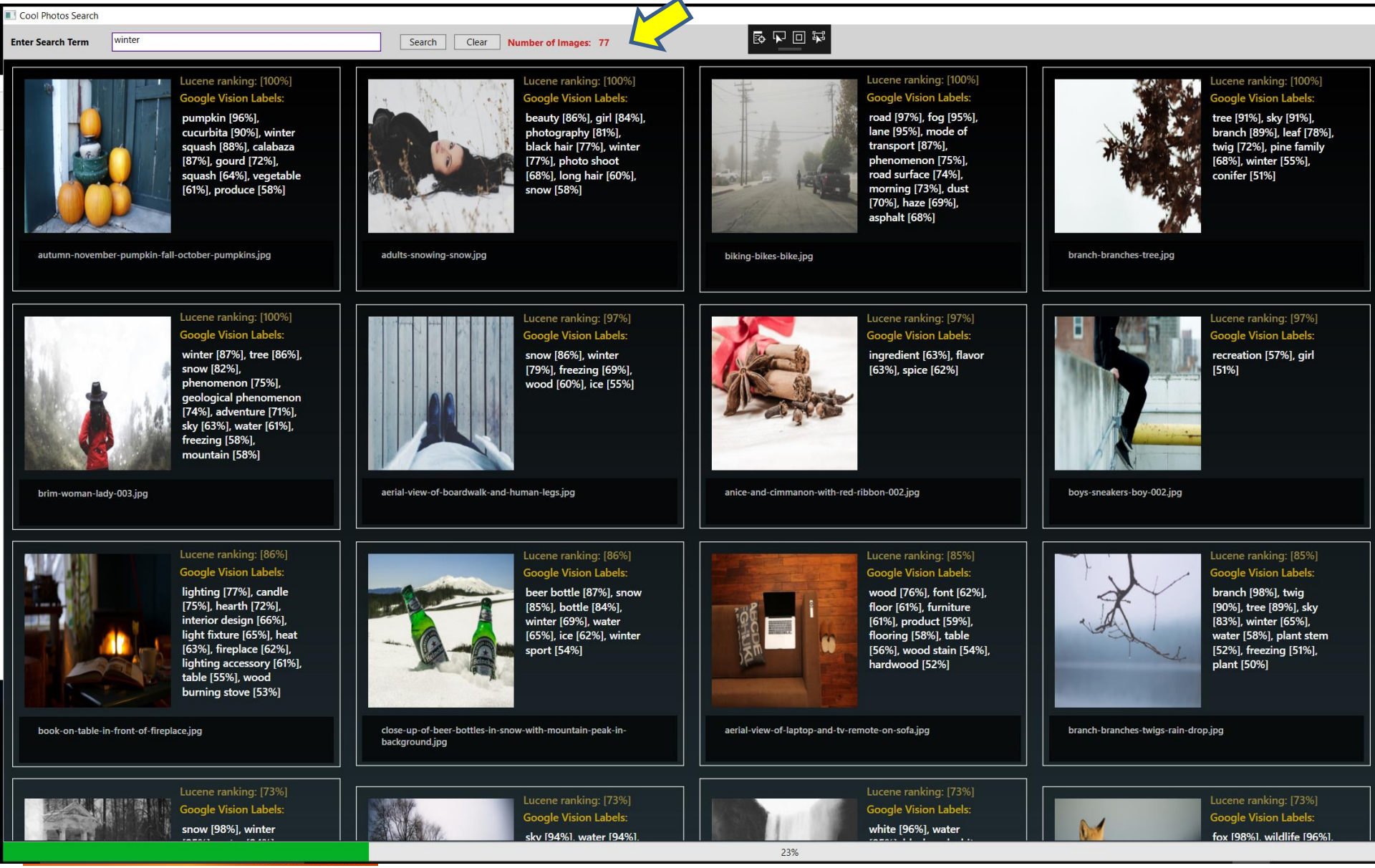
Clear will bring you back to Main menu and Clear search query example

Search results for “sea” query - 96 hits in the example)



System snapshots – Search (winter)

Search results for “winter” query - 77 hits in the example)



Installation options

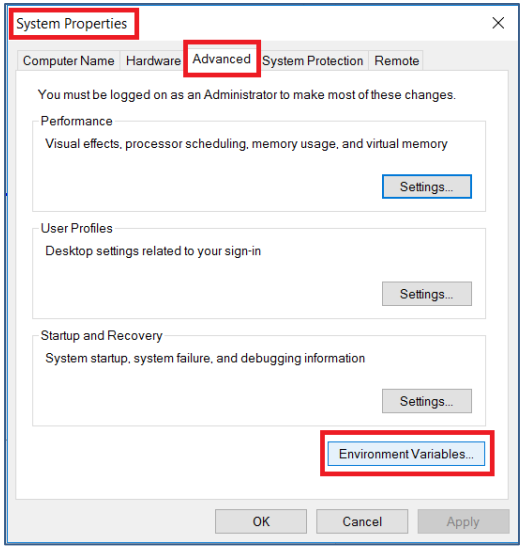
Link to Installer Download (Windows only!)

- https://www.dropbox.com/s/ef38sas1o7vxji8/CoolPhotoSearchSetup.msi?dl=0

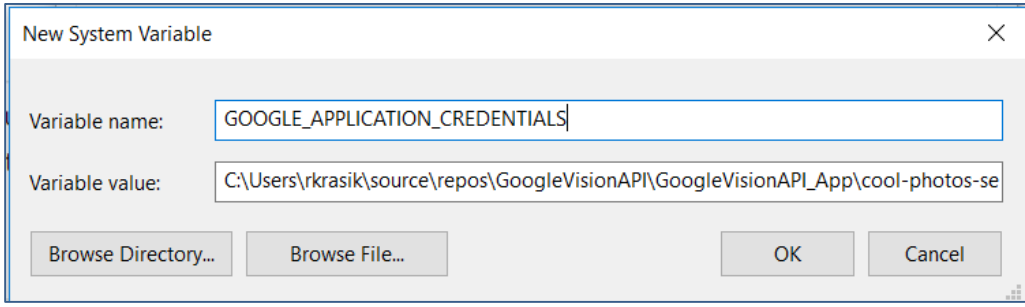
Link to GitHub (Source Files)

- https://github.com/rkrasik7/cs_410_final_project_photos_search
- In order to compile and run in **Visual Studio 2017** - there is need to uninstall the Installer (in case it was installed) and define relevant System Environment variables

STEP 1



STEP 2



Variable name: GOOGLE_APPLICATION_CREDENTIALS
Variable value (e.g. depends on the Git directory):
C:\Users\rkrasik\source\repos\GoogleVisionAPI\GoogleVisionAPI_App\cool-photos-search-57852b03daf7.json

STEP 3

Key learnings

- Performance is challenging when you are dealing with pictures - need to consider resizing, optimizations, multithreading etc.
- Pictures validation is essential, you might face the same picture but in a different quality
- Nothing is for free - my 300$ Google Cloud Platform budget ran out within 2 weeks☹
- The Google Vision and Lucene APIs are quite rich, yet sometime not as comprehensive with regards documentation…

Further extension

- Add additional source inputs
  - User laptop / USB drive / Google Drive / Google Photos / Instagram / Facebook / …
- Add complex searches
  - AND / OR / IN operators support
- Add operations for search results
  - Save picture for single picture
  - Select/multi-select pictures and then “Save pictures”
  - Save query
- Further performance boost
- User Experience and Usability enhancements


This is
THE END
and
Thank you
For watching