SE 3XA3: Software Requirements Specification
google-images-downloader

Team 201, CAS Dream Team
Sam Crawford, crawfs1, 400129435
Joshua Guinness, guinnesj, 400134735
Nicholas Mari, marin, 400132494

February 8, 2020

Contents

List of Tables

List of Figures

Table 1: **Revision History**

Date	Version	Notes
1/21/2020	1.0	Added to Repo
1/28/2020	1.1	Filled in name of project and team
2/08/2020	1.2	Initial draft of SRS

This document describes the requirements for google-images-downloader. The template for the Software Requirements Specification (SRS) is a subset of the Volere template (**?**). If you make further modifications to the template, you should explicitly state what modifications were made.

1 Project Drivers

1.1 The Purpose of the Project

The purpose of this project is to re-create an open source software application following the software development life cycle, for the software engineering course 3XA3. This project will give the group exposure and experience in how to properly develop a software product from beginning to end.

The project chosen is a google images downloader command line tool that will allow end users to download a certain number of google images given keywords. We want this product primarily to be able to help those involved in machine learning, and secondarily those involved in art.

1.2 The Stakeholders

The following subsections will discuss various stakeholders in the product being developed.

1.2.1 The Client

Dr. Asghar Bokhari, the professor of our course, and the TA responsible for marking all our deliverables, Andrew Lucentini.

1.2.2 The Customers

There are two main types of customers who would use the product, those involved in machine learning and those in art.

Individuals, or companies who are using machine learning algorithms or neural networks for image recognition purposes could benefit from this product as it would allow them to quickly obtain a large variety of images on a specific keyword. This can help them make their models more accurate and robust.

Artists may use this tool for projects requiring many images based off a single keyword such as collages or reference work.

1.2.3 Other Stakeholders

Other stakeholders include Google Images because their service is being used to scrape images off of it.

1.3 Mandated Constraints

1.3.1 Schedule Constraints

The project deliverables must be met on time according to the project description. The remaining ones are included below.

- Proof of Concept Demonstration - February 11, 2020
- Test Plan Revision 0 - February 28, 2020
- Design & Document Revision 0 - March 13, 2020
- Revision 0 Demonstration - March 17, 2020
- Final Demonstration (Revision 1) - March 31, 2020
- Final Documentation (Revision 1) - April 6, 2020

Further breakdown of these deliverables their respective group member assignments can be found in our Gantt Chart.

1.3.2 Solution Constraints

Description: The product shall be developed in Python 3. Rationale: The existing implementation of the software is in Python 3, making it easier to re-develop if it is in the same language. Fit criterion: The software is programmed in Python 3.

Description: The product shall get the image URLs from Google Images by scraping the HTML code returned. Rationale: The existing implementation of the software designed its solution in this was so since we are re-developing it, that component of the solution will remain the same. Fit criterion: The application works by scraping HTML code to get URLs

1.3.3 Budget Constraints

The budget for this project is $0 so the software must be able to built without spending any money.

1.3.4 Off-the-Shelf Software

Since the purpose of the project is to re-implement an existing open source project, the existing version of the software will be reference heavily when developing this version. The existing software can be found at https://github.com/hardikvasa/google-images-download.

In addition to some common python libraries, urllib, urllib2, & httplib will be used to manage and navigate URLs in the application.

1.3.5 Partner or Collaborative Applications

The software product being developed will use the Google Images application to download images onto the local computer. The application will use the image URL links that are embedded in the HTML code for the image returned in a Google Images search query. These URLs are stored in 'data-iurl' under the 'img' tag.

1.3.6 Implementation Environment of the Current System

The software product being developed will be installed on users local machines, or on servers, to function as a command line tool.

1.4 Naming Conventions and Terminology

Table 2: **Naming Conventions and Terminology**

Name/Term	Definition
The System	The product defined in this document
HTML	Hypertext Markup Language
URL	The address of a website or file on the internet
Python 3	The third version of the python programming language
Keyword	The search term used to find images
Limit	The number of images the user wished to download
Google Images	The image search engine provided by Google
Search Constraints	restrictions images must follow such as size or file type
JPEG	A format for image files

1.5 Relevant Facts and Assumptions

It is assumed that users of the application have basic computer skills and are able to type commands on a command line given examples. It is also being assumed that users have basic knowledge of pictures, like what aspect ratio means, or colour type refers to.

2 Functional Requirements

2.1 The Scope of the Work and the Product

2.1.1 The Context of the Work

In order for the product to function as intended, the system must interact with Google Images. As a result there is a need to understand how Google Images works. This is because the original implementation of this product scrapes the HTML off of Google Images and then finds the links to the original images in the scraped HTML. Therefore, the specifics of what is required background knowledge for this product is how images are stored on the website so they can be effectively found in the scraped HTML.

2.1.2 Work Partitioning

BE1: The User Requests an Image.

Inputs:

- Keyword
- Limit
- Search Constraints

Outputs:

- Images downloaded to computer

Summary:

- Download a number of images equal to the limit of the given keyword that also match any other given search constraints.

2.1.3 Individual Product Use Cases

The main use case of this product is the user actor requesting an image. This use case can include the user specifying optional input arguments such as a limit of images, or other search constraints such as prefixes for searching or the size of images wanted.

The following scenarios will outline a general interaction between a user actor and the system for both the main use case, and sub use cases.

Scenario 1: The user requests an image without specifying a keyword. The system will return a message with the proper format for an image request and remind the user that a keyword is required.

Scenario 2: The user requests an image with a specified keyword. The system will retrieve images from Google Images that match the specified keyword. Since a limit on images was not specified the system will download the 100 images matching the search criteria. Since an output directory was not specified, the system will download the images into a downloads directory located in the same directory where the program was executed.

Scenario 3: The user requests a limit of 50 images with the keyword "Cat". The system will retrieve the first 50 images off of Google Images

when Cat is used as the search term. Since an output directory was not specified, the system will download the images into a downloads directory located in the same directory where the program was executed.

Scenario 4: The user requests a limit of 50 images with the keyword "Cat". The user specifies a download directory of "./Images/Cat/". The system will retrieve the first 50 images off of Google Images when Cat is used as the search term. Since an output directory was specified, the system will create a new directory named images then inside images create a new directory called cats. The images will then be downloaded into the new cats directory. If the images and cats directory already exists, the system will use the existing directory rather than creating a new one.

Scenario 5: The user requests a limit of 100 images with a specified keyword and download directory. The user also specifies specific search constraints. The system will search with the keyword and download 100 images that match the keyword and don't violate any of the search constraints. For example if the search constraint is a file type of JPEG, the system will only download JPEG images. The images will be downloaded into the specified directory, creating it if it does not already exist.

2.2 Functional Requirements

FR 1: The user shall be able to download images that match a given keyword.

Rationale: This is the main functionality of the product. With this, users will be able to download multiple images that all match the keyword given with just one command.

FR 2: The user shall be able to specify the number of images they want

Rationale: This allows the user to select how many images they want and will be a useful quality of life feature when the user knows approximately how many images they want.

3 Non-functional Requirements

3.1 Look and Feel Requirements

3.2 Usability and Humanity Requirements

3.3 Performance Requirements

3.4 Operational and Environmental Requirements

3.5 Maintainability and Support Requirements

3.6 Security Requirements

3.7 Cultural Requirements

3.8 Legal Requirements

3.9 Health and Safety Requirements

This section is not in the original Volere template, but health and safety are issues that should be considered for every engineering project.

4 Project Issues

4.1 Open Issues

4.2 Off-the-Shelf Solutions

4.3 New Problems

4.4 Tasks

4.5 Migration to the New Product

4.6 Risks

4.7 Costs

4.8 User Documentation and Training

4.9 Waiting Room

4.10 Ideas for Solutions

5 Appendix

This section has been added to the Volere template. This is where you can place additional information.

5.1 Symbolic Parameters

The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance.