Software Requirements Specification

for

Aerolyzer

Version 1.0 approved

Prepared by E. Reilly Collins, Sophia Liu, and Jesse Hanson

Group 22

CS 461 Fall 2016

November 7, 2016

Abstract: This document contains the software requirements specification for Aerolyzer. Aerolyzer is a mobile Web application capable of processing visible images and inferring atmospheric phenomena to provide the general public with near-real time monitoring of aerosol conditions. The goal of this document is to clearly specify the requirements for our mobile app to be developed.

CONTENTS

I. Introduction

A. Purpose

The purpose of this project, Aerolyzer (version 1.0), is to develop a mobile web application that supports the ability to understand content from user-uploaded digital outdoor images. Our goal is to allow the general public to better monitor current aerosol content.

B. Scope

This software will be a mobile web application for use by the general public. This application will be designed to uniquely utilize color distribution within an uploaded mobile image to identify features necessary for analyzing aerosol content. The end product will be a full mobile web app that employs an open source algorithm translating how the human brain interprets images using RGB values. Citizens in the general public will be able to quantify color in images and use the resulting data to determine current aerosol content.

More specifically, users will be able to upload an image of an outdoor scene taken on a mobile phone. The application will then give output detailing the current aerosol content of the atmosphere based on the uploaded image. This software will provide average citizens with near-real time monitoring of atmospheric conditions.

C. Definitions, acronyms, and abbreviations

Aerosol: Minute particles suspended in the atmosphere. When these particles are sufficiently large, we notice their presence as they scatter and absorb sunlight. Their scattering of sunlight can reduce visibility (haze) and redden sunrises and sunsets. [1]

EXIF/exif: Exchangeable image file format specifying the specs of digital/smartphone cameras.

MISR: Multi-angle Imaging SpectroRadiometer satellite. [3]

More definitions, acronyms, and abbreviations will be added as needed to this requirements specification document.

D. References

[1] Bob Allen. (1996) Atmospheric Aerosols: What Are They, and Why Are They So Important? [Online]. Available: http://www.nasa.gov/centers/langley/news/factsheets/Aerosols.html

[2] WeatherAPI: Introductions. [Online]. Available: https://www.wunderground.com/weather/api/d/docs

[3] MISR: Home Page. [Online]. Available: http://www-misr.jpl.nasa.gov/

E. Overview

The next section, Overall Description, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next section. The third section, Specific Requirements, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

II. Overall Description

This section will give an overview of the whole application. The app will be explained in its context to show how it interacts within a larger system, as well as introduce basic functionality. It will also describe the type of stakeholders who will use the app and what functionality is available for them. At last, the constraints and assumptions for the system will be presented.

A. Product Perspective

Aerolyzer will be an independent and self-contained mobile web application. It is not designed to be a component of a larger system. Mobile images and associated metadata used for the color-analyzing algorithm will be from various online sources, such as Weather Underground.

As a note, Aerolyzer will be similar to other citizen science applications, though not part of those applications, by allowing user images to be used in improving the machine learning of the algorithm. More information about the citizen science component of the system will be added as needed to this requirements specification document.

B. Product Functions

Within the mobile application, users will be able to upload an image of an outdoor scene with the sky in view from their phones camera roll. An algorithm will use the image's colors and associated metadata, combined with other meteorological data, to output current atmospheric conditions and aerosol content. Output from the color-analyzing algorithm will be displayed on the screen for the user.

C. User Characteristics

1) Users will be smartphone owners.
2) Users will need technical knowledge about the basic functionality of their smartphones to take photos.
3) Users can access the application on their desktop computer or smartphone, but will need to know how to use a web browser and navigate to the Aerolyzer app for both interfaces.
4) Users will need to have access to mobile photos whether they are on the smartphone or desktop version of the web application.
5) Users educational levels can be anywhere across the board; they do not need expertise in aerosols in order to understand output.

D. Constraints

1) Although the system is a web application, uploaded images must be taken on phones in order for the algorithm to give correct output. Images from mobile phones include associated metadata that will be needed for the aerosol analyzing algorithm, and this metadata is not present on non-mobile cameras.
2) Images uploaded to Aerolyzer cannot be cleaned up before being uploaded, as this could distort the color. This means no filters or other image editing tools can be used on the images.
3) Only direct landscape images with the sky in view can be uploaded to Aerolyzer. For example, images taken from inside a window or a picture of grass will not give accurate output about current aerosol conditions.

More information about constraints of the system will be added as needed to this requirements specification document.

E. Assumptions and Dependencies

1) We assume users are willing to give permission for having their images accessed and processed.
2) We assume that users have access to at least one mobile image in their photo library to be uploaded.
3) We assume that users are connected to the Internet.
4) We assume users have access to a web browser.

III. Specific Requirements

This section will give an overview of specific requirements for the application. Links to external systems will be explained to show what other resources the application will use. The app's functions in terms of individual use cases will be described in detail.

A. External Interfaces

One link to an external system will be accessing weather information from the Weather Underground database using their Weather API. We will need this to gather astronomy data, almanac information, and current conditions (at a minimum). The output from this API will be used as input by the algorithm that determines current aerosol content. [2]
Another link to an external database will be pulling outdoor images from online sources to be used in the color-analyzing algorithms machine learning. The source of these images will be from reputable repositories of images that include associated mobile camera metadata, such as Weather Underground. The purpose of these images will be to use them as input in order to improve the algorithms machine learning component.
A final link to an external interface will be the access to stored images and associated output from the color-analyzing algorithm. More information about image and data storage will be added as needed to this requirements specification document.

B. Functional Requirements

1) Web Application Interface:

Trigger: Users access the Aerolyzer mobile web application.

Precondition: Users are on a mobile device with access to a web browser and the Internet.

Basic Path:

1) The user opens a Web browser on his or her mobile device.
2) The user navigates to the Aerolyzer URL using the web browser.
3) The system shall load the start page on the users web browser.

Alternative Paths: Users can access the system by clicking a link that opens up their mobile web browser.

Postcondition: The user has access to the Aerolyzer mobile web application.

Exception Paths: If the user does not have Internet access, the use case is abandoned.
If the user is trying to access the system from a desktop device, the system alerts the user that only mobile images are allowed and the use case is abandoned.

2) Upload Photo:

Trigger: User selects button to upload a photo to be analyzed.

Precondition: Users are on a mobile device with at least one image in its photo library.

The web browser used to access the system has access to the users mobile photos.

Basic Path:

1) The user selects the Upload a Photo option from the systems start page.
2) The mobile devices photo library is displayed with the option to select one photo to be uploaded.
3) The user selects one photo to be uploaded and then selects the Use Photo button.

Alternative Paths: None.

Postcondition: The user is taken to a loading page while the photo is being analyzed.

Exception Paths: In step 3., the user may abandon the upload at any time by selecting the Cancel option.

If the photo being uploaded does not meet constraints identified in section 2.4, the image cannot be used as input for the color-analyzing algorithm. The user will be notified and the use case is abandoned.

3) Display of Aerosol Content:

Trigger: Photo has been analyzed by the algorithm and the resulting aerosol content has been determined.

Precondition: A photo has been successfully uploaded as per use case 2.

Basic Path:

1) A loading screen is displayed and the user waits for the photo to be analyzed.
2) The photo is used as input for the color-analyzing algorithm.
3) The algorithm outputs data for current atmospheric conditions.

Alternative Paths: None.

Postcondition: The user is taken to a page where the resulting aerosol content of the photo uploaded is displayed.

Exception Paths: In step 1., the user may abandon the analysis at any time by selecting the Cancel option.

4) Extract EXIF Information From Data:

Trigger: A script that extracts EXIF data from an uploaded image is run.

Precondition: A photo has been successfully uploaded as per use case 2.

Basic Path:

1) The Aerolyzer developer receives images as input for the script.
2) The script outputs a JSON file containing the filename as a string, the category of/tags for the photo (e.g. clouds, fall colors, sunrise, etc.) as a list, and the EXIF data associated with the image as a dictionary.
3) The JSON file retrieved in step 2 is used as input for the core color-analyzing algorithm and is stored in the central storing location.

Alternative Paths: Images and associated metadata from any source can be manually added to the central storing location.

Postcondition: Aerolyzer's image repository includes retrieved images for use in improving the color-analyzing algorithm.

Exception Paths: None.

5) Using Weatherunderground API:

Trigger: Picture is uploaded and ready to be analyzed.

Precondition: A photo has been successfully uploaded as per use case 2.

Basic Path:

1) Meteorological and astronomical data is retrieved from the Weather Underground API.
2) The information derived from the API is given as input to the color-analyzing algorithm.
3) The uploaded image is analyzed via the algorithm.

Alternative Paths: None.

Postcondition: The color-analyzing algorithm runs using Weather Underground API data and the user is taken to a page where the resulting weather information based on the photo uploaded is displayed.

Exception Paths: If data is unable to be retrieved from the Weather Underground API, this data will not be used as input for the color-analyzing algorithm and may affect the accuracy of results.

If data is unable to be retrieved from the Weather Underground API, this data will not be displayed to the user when viewing aerosol content results.

6) Get Data From Satellite Source:

Trigger: Picture is uploaded and ready to be analyzed.

Precondition: A photo has been successfully uploaded as per use case 2.

Basic Path:

1) Geo-related data is retrieved from the satellite source (such as MISR) based on the iamge's location.
2) The information derived from the satellite source is given as input to the color-analyzing algorithm.
3) The uploaded image is analyzed via the algorithm.

Alternative Paths: None.

Postcondition: The color-analyzing algorithm runs using satellite data and the user is taken to a page where the resulting satellite information based on the photo uploaded is displayed.

Exception Paths: If data is unable to be retrieved from the satellite, this data will not be used as input for the color-analyzing algorithm and may affect the accuracy of results.

If data is unable to be retrieved from the satellite, this data will not be displayed to the user when viewing aerosol content results.

7) Running the Core Algorithm:

Trigger: Picture is uploaded and ready to be analyzed.

Precondition: A photo has been successfully uploaded as per use case 2.

EXIF, meteorological/astronomical, and geo-related data has been successfully extracted as per use cases 4, 5, and 6, respectively. **Basic Path:**

1) The application uses EXIF, meteorological/astronomical, and geo-related data as input for the core color-analyzing algorithm.
2) The core algorithm runs.

Alternative Paths: None.

Postcondition: The output from the algorithm is used to display aerosol content as per use case 3 and is stored in the central storing location.

Exception Paths: If the needed meteorological data is not available, the algorithm will still run and give output that does not include the meteorological data.

C. Performance Requirements

The web application will have both mobile-scaled and desktop-scaled interfaces.

The system will be able to analyze at most 500 images per day total and at most 10 images per minute total (meaning for all combined users, not each individual user).

More performance requirements will be added to this requirements specification document after further research on the color-analyzing algorithm capabilities is complete.

SIGNATURES

Kim Whitehall

Client, NASA JPL

Signature: _____

Date: November 7, 2016

E. Reilly Collins

Student, Oregon State

Signature: _____

Date: November 7, 2016

Sophia Liu

Student, Oregon State

Signature: _____

Date: November 7, 2016

Jesse Hanson

Student, Oregon State

Signature: _____

Date: November 7, 2016