Software Design Specification

for

Photo Metadata Extractor Tool

Version 1.1

Prepared by Bryan Portillo, Evan Putnam, Kay Vargas

Cal Poly Humboldt

10/01/2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
First Daft	09/27/23	Initial document. Setup headers for table of contents.	1.0
Populating Content	10/01/23	Adding content to the sections of the design document	1.1
Updating Functional Requirements	10/04/23	Added specific documentation relating to dynamic behavior satisfying functional requirements	1.2

1. Introduction

1.1. Purpose

The Photo Metadata Extractor Tool (PMET) is designed to identify physical book and text information such as Superintendent of Documents (SuDoc) numbers, title, and publication data through the analysis of inputted images of the front page of documents. This software then returns the collected data to the user in a formatted csv file. Revision 1.0

1.2. Statement of Scope

This document serves to explain the internal data design, architecture, user interface, restrictions, limitations, and testing procedure of the PMET software program. This document will provide an in-depth overview of these structures in order to allow the reader to better understand the underlying mechanics of this software. This software is ideally suited for libraries aiming to generate data on large sets of documents in a streamlined and effective manner and is not intended for a more broad population.

1.3. Software Context

This software contains three major components consisting of a machine learning model (MLM) for text recognition within images, an API based query system, and an interactive user interface designed for user interaction with the other two components. Each of the major components are interdependent on each other and must be functional in order for the software to be used.

1.4. Major Constraints

The major constraint of this software is its capacity to successfully and accurately classify documents through their SuDoc. While the software is designed to optimize performance along with the machine learning model's performance there seems to be a strong upper bound to the potential accuracy of the software due to the variability in text documentation in external databases and the different library dependent formatting of SuDoc numbers on scanned documents.

2. Data Design

2.1. Internal Software Data Structure

The PMET software does not retain any data after any specific interaction with the user. This software will read image data from a user specified folder which will then be fed through a data-pipeline. Upon completion of the data pipeline the generated data file will be deleted once the user has selected to conclude the query or the program times out.

2.2. Global Data Structure

No global data structure was utilized for this software.

2.3. Temporary Data Structure

No temporary data structure was utilized for this software.

2.4. Data Pipeline

The date pipeline consists of seven total steps which are described and labeled in the following paragraph. A graphical representation is available in appendix B.

The PMET software will take image[s] as input from the user (1). These images will be accessed from a user specified file on their local machine which will be accessed throughout the recognition and classification process. Images will then be converted into tensor and then fed into the machine learning model which will then extract identified text from the images (2). The extracted text will then be run through a post processing algorithm (TBD 1)(3). After post-processing the resulting text will be fed into the software query system which will then query the WorldCat Search API v. 2 for the document title and publication year of the corresponding document[s](4). In the case of a failed query the user will then be prompted to verify the given data points through the original inputted image(5). Once the query process is complete the extracted data will then be written into a csv file which the user will be provided with the option to save(6). Once the user has hit the 'complete query' button or the software times out the csv file will be automatically deleted(7).

3. Architectural and Component-Level Design

3.1. System Structure

The system will consist of three components:

- Interactive User Interface
- Machine Learning Model
- Database Query System

3.2. Description for Interactive User Interface

This interface forms a bridge for the user to interact with and access the other two components within the PMET software.

A user will be able to toggle or view the following actions and process:

- Login
- Select file location
- Begin query

- View query process status and estimated remaining time
- Cancel/Conclude query
- Download the resulting output
- Verify and correct failed queries
- Preview resulting classification

3.3. Dynamic Behavior for Interactive User Interface

1. Select File Location: The user will toggle a folder to pull document images from. If the file is empty or contains no images files the user will receive an error message and be asked to reselect a folder.

   This will meet the following requirement[s] for the MLM

   - O_REQ-4: If the data inputted by the user is not acceptable for the model the users will be notified and the data will not be processed.

2. Begin Query: Once the user has selected to begin the query the interface will then ping the user through a file selection process in which the folder containing the to-be-processed images will be selected

   This will meet the following requirement[s] for the IPI

   - I_REQ-1 - The IPI will offer an input field which allows the user to load batches or individual document images which can then be processed by the program.
   - I_REQ-2 - The IPI will be able to accept images in many formats

3. Login: The user will be prompted to login using a client ID and a client secret which will ultimately be provided from the underlying API and must be requested by the software user independently in order to use the software. Once the user clicks 'login' with their login credentials a query will be generated to WorldCat Search API v. 2. If the query is successful the user will be taken to the homepage of the interface, otherwise they will be prompted to resubmit their login credentials.

4. Cancel/Conclude query: Upon the user clicking this button the program will cancel the query and delete any intermediate or output data. The interface will then return to the homepage allowing the user to instantiate a new query if requested.

   This will meet the following requirement[s] for the IPI

   - I_REQ-7 - The user will be able to cancel the query or stop the process if they choose to

5. Download the Resulting Output: Upon selecting this option the output csv file will then be downloaded onto the users local machine. The output csv will be cleanly formatted in a user friendly manner.

   This will meet the following requirement[s] for the IPI

- I_REQ-5 - The user will be easily able to download the resulting data at the end of the entire process
- I_REQ-3 - The IPI will Output the results of searching the database in a formatted file which is easily accessible to the user and in the specified form csv with labeled data points.

6. Verify Correct and Failed Queries: During the query process the user may be pinged to verify the extracted text. In such a case the user will have the opportunity to fill in the resulting text fields and view the originally imputed document image. Upon completion the user will hit the 'complete verification' button and the query will be resubmitted.
   This will meet the following software requirement[s] for the MLM and IPI:
   - O_REQ-4: If the data inputted by the user is not acceptable for the model the users will be notified and the data will not be processed.
   - I_REQ-4: When a database search is unsuccessful the IPI will notify the user and ping them to verify the recognized SuDoc from the text. In the case the search fails after verification the user will then be notified and prompted to manually fill in the needed data fields.
   - D_REQ-5: The system will ping the user for verification in the case of a failed query. The user will be able to verify the SuDoc and re-initiate the program to query the database again with the verified SuDoc in the case of a correction.

7. Preview Resulting Classification: At the completion of the query the user will be pinged with an option to preview their outputted classification text. Upon selecting yes the top ten or entire [depending on the total file amount input] rows of the file will be displayed within the interface.
   This will meet the following software requirement[s] for the IPI and DQS:
   - I_REQ-3 - The IPI will Output the results of searching the database in a formatted file which is easily accessible to the user and in the specified form csv with labeled data points.
   - D_REQ-6: Data must be formatted into an easily readable file and return the file to the interactive program interface

8. While the query is in progress there will be a displayed bar denoting the estimated time remaining for processing.
   This will meet the following software requirement[s] for the IPI:
   - I_REQ-6 - The user will be able to determine the current state of their query

3.4. Description for Machine Learning Model

This machine learning model serves as the text classifier and processor from all inputted documents images from the user. The user will not be able to directly interact with the model except through the interactive user interface. However the user will be able to view the total time remaining and the current status of the model in processing the inputted data.

3.5. Dynamic Behavior for Machine Learning Model

1. Once the user has selected to begin a query the machine learning model will automatically and autonomously run through the steps (2) and (3) of the data pipeline (Sec 2.4). [View appendix B]

   This will meet the following requirements for the MLM:

   - O_REQ-1: Inputted user data must be formatted into an acceptable format for the machine learning model to be able to successfully process the data
   - O_REQ-2: The underlying model must be able to identify and extract image text from the images inputted by the user.
   - O_REQ-3: The identifying text must be parsed text into recognizable categories such as SuDoc, title, publish date, etc….

3.6. Description for Database Query System

The database query system will utilize the WorldCat Search API v. 2 to query the selected registry (TBD 2). The user will only interact with the system through the interactive user interface. After a query has been instantiated the program will run through the processes of this component autonomously and the user will only interact with this component in the case of a failed query.

3.7. Dynamic Behavior for Database Query System

1. Once a query has been instantiated and the steps (2) and (3) of the data pipeline have been completed. The database query system will then initialize and complete steps (4) and (5).

   This will meet the following requirement[s] for the DQS:

   - D_REQ-1: The database query system will be able to connect to WorldCat Search API v. 2.
   - D_REQ-2: The query system will be able to use the SuDoc to find the corresponding catalog entry in the system for the given document.
   - D_REQ-3: The query system will extract specific data points from the query and write them to a secondary file.

2. Step (5) will only be triggered in the case of a failed query. In such a case the user will then re-verify the data (Sec 3.3.4) and upon completion the query will be reinstated. If this second query fails the user will then be notified however the query will not be requested a second time.

   This will meet the following requirement[s] for the DQS:

- D_REQ-4: The system will notify the user in the case the query fails even after the user has verified the SuDoc and automatically populate data fields with anticipated data based on the machine learning model text parsing.
- D_REQ-5: The system will ping the user for verification in the case of a failed query. The user will be able to verify the SuDoc and re-initiate the program to query the database again with the verified SuDoc in the case of a correction.

4. User Interface Design

4.1. Description of the User Interface

The user interface consists of several states: login, directory selection, SuDoc verification, query manipulation, output review, software settings, and bug reporting. The interface prioritizes function and readability over aesthetic. This interface will serve a pivotal purpose for the software by serving as the central unit connecting all of the software components and enabling user interaction within the software.

4.2. Interface Design Rules

The interface does not require complicated functionality so it will inherently be quite simple and straightforward.

- Ease of use
- Simplicity
- Interactive design
- Accessibility
- Consistency
- Self-driven

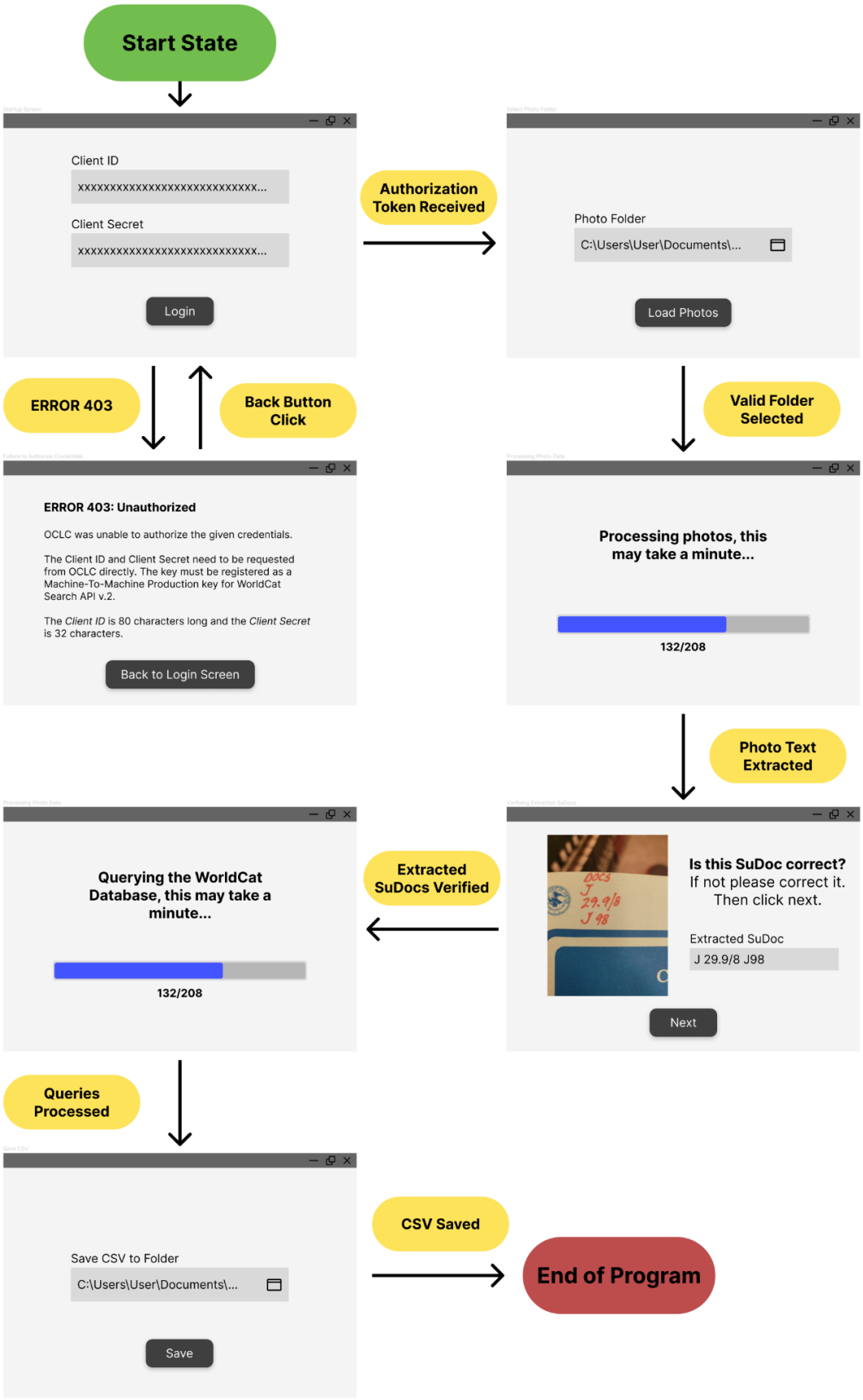
4.3. Components Available

Reference Section 3.2 for a detailed description of the interfaces components and processes.

4.4. UIDS Description

No UIDS was utilized for this software.

4.5. Interactive User Interface Storyboard



5. Restrictions, Limitations, and Constraints

5.1. LIST OF RESTRICTIONS OR OTHER

5.1.1. Database (TBD 2) accuracy is a limitation as not every document query will have a record and thus query success will be limited by the extent to which documents are recorded within the database.

5.1.2. The SuDoc classification on documents varies by year and is sometimes in typeset or handwritten form with varying punctuation. While the model will be trained for optimizing performance its accuracy may suffer for fringe cases.

5.1.3 If the user inputs document images with large amounts of text the softwares processing time will be impacted. Thus in the case of exceptional runtimes the user should limit imputed document images to minimize total text.

6. Testing Issues

6.1. Classes of Tests for Machine Learning Model

- Text extraction accuracy
- Text categorization accuracy
- Verification of user selected data
- Robustness of distinguishable data input instances

6.2. Expected Software Response for Machine Learning Model

Note: View the Glossary for an in-depth description of each requirement ID category

Functional Area	Requirement ID	Test Data/ Input	Expected Output	Actual Response
MLM	O_REQ-1	Single document image within a folder	Extracted text vector	
MLM	O_REQ-1	Multiple document images within a folder	Multi-dimensional matrix of columns corresponding to extracted text vectors	
MLM	O_REQ-4	Empty folder	Error code (TBD 3)	
MLM	O_REQ-4	Folder containing no	Error code (TBD 3)	

		image files		
MLM	O_REQ-2	Document images with text and no images on cover	Text vector containing at least 90% of text with accurate punctuation	
MLM	O_REQ-2	Document images with text and cover images	Text vector containing at least 90% of text with accurate punctuation	
MLM	O_REQ-2	Document image with no text	Empty text vector	
MLM	O_REQ-3	Document images	Text vector will be organized correctly into distinguishable categories with a minimum accuracy rate per word of 90%	
MLM	O_REQ-4	Single image of non-image format [no selected folder]	Error code (TBD 3)	

6.3. Performance Bounds for Machine Learning Model

Performance for the machine learning model will be measured through the following two constraints.

- Runtime: This will be defined as the total time taken to process and extract text from inputted images. Optimal runtime should not exceed more than two seconds for a single image.
- Accuracy: This will be defined as the percentage of correctly classified characters from an inputted image.

6.4. Classes of Tests for Interactive User Interface

TBD(3)

6.5. Expected Software Response for Interactive User Interface

TBD(4)

6.6. Performance Bounds for Interactive User Interface

TBD(5)

6.7. Classes of Tests for Database Query System

TBD(6)

6.8. Expected Software Response for Database Query System

TBD(7)

6.9. Performance Bounds for Database Query System

- Success Rate: This will be defined as the total percentage of queries which successfully return data corresponding to the inputted image texts
- Query Time: Query time will represent the total time it takes to complete a single query including the time involving user verification in the case of an initially unsuccessful query

6.10. Identification of Critical Components

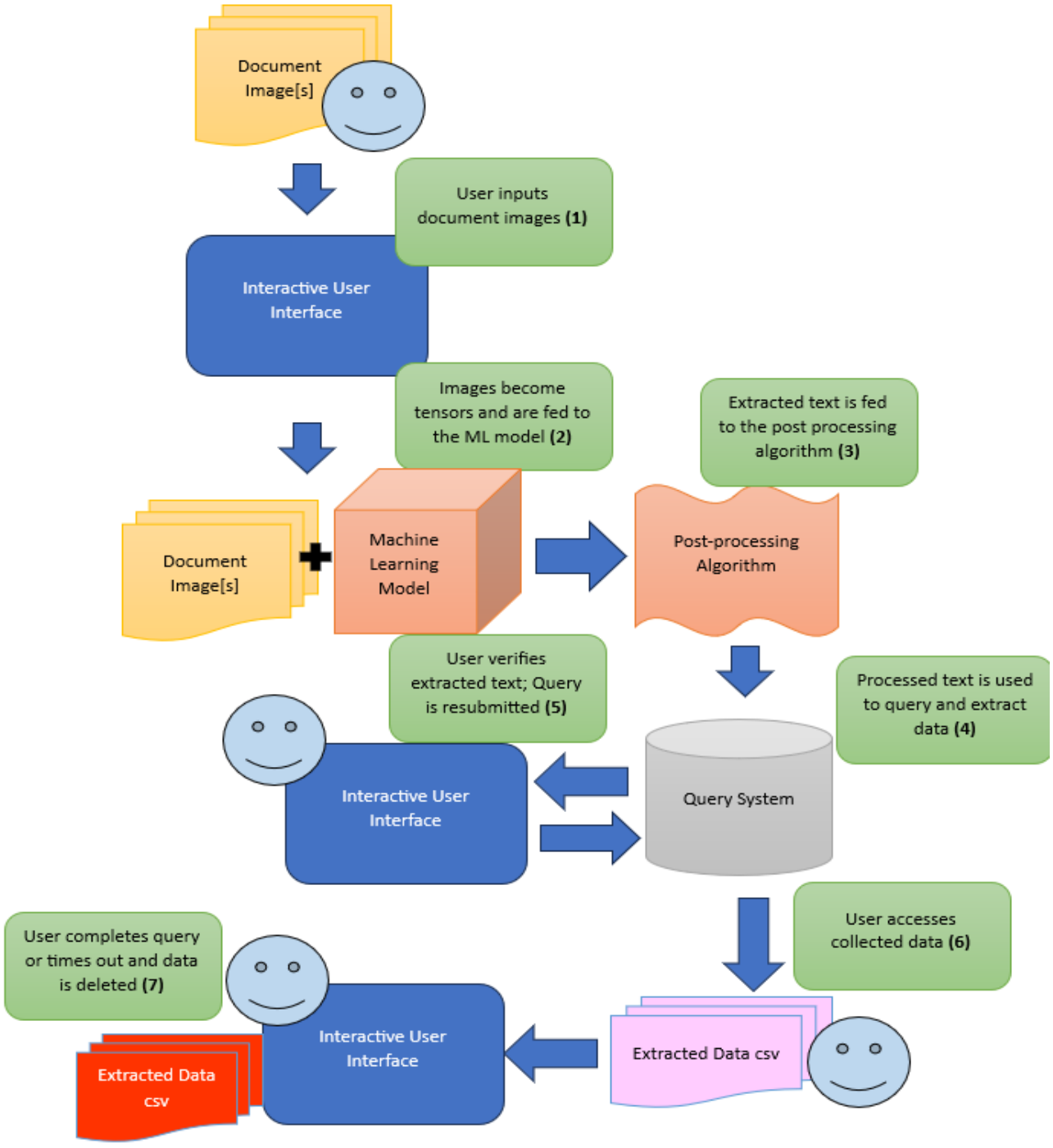
The critical components of PMET are the machine learning model, interactive user interface, and the database query system. Each of these components are necessary to generate a record from a photo of a document and the system will be unable to function unless each of these pieces are working cohesively.

Appendix A: Glossary

PMET - Photo Metadata Extractor Tool
API - Application Programming Interface
MLM - Machine Learning Model
IPI - Interactive Program Interface
DQS - Database Query System

Appendix B: Analysis Models

Data Flow Diagram



Appendix C: To Be Determined List

TBD(1): Post processing steps to be determined and outlined at a later time
TBD(2): Registry to be determined
TBD(3): Specific error code for empty input error for MLM
TBD(4): Test Classes for interactive user interface
TBD(5): Expected software response for interactive user interface
TBD(6): Performance bounds for interactive user interface
TBD(7): Classes of tests for database query system
TBD(8): Expected software response for database query