OCR Based Document Archiving and Indexing Using PyTesseract: A Record Management System for DSWD Caraga, Philippines

Conference Paper · December 2020

DOI: 10.1109/HNICEM51456.2020.9400000

Jaymer Jayoma
Caraga State University
11 PUBLICATIONS 26 CITATIONS
SEE PROFILE

Elbert Moyon
Caraga State University
6 PUBLICATIONS 11 CITATIONS
SEE PROFILE

Edsel Matt Otacan Morales
Caraga State University
13 PUBLICATIONS 29 CITATIONS
SEE PROFILE

OCR based Document Archiving and Indexing using PyTesseract: A Record Management System for DSWD Caraga, Philippines

Jaymer M. Jayoma
College of Computing and Information Sciences (CCIS)
Caraga State University
Ampayon, Butuan City, Philippines, 8600
jmjayoma@carsu.edu.ph

Elbert S. Moyon
College of Computing and Information Sciences
Caraga State University
Ampayon, Butuan City, Philippines, 8600
esmoyon@carsu.edu.ph

Edsel Matt O. Morales
College of Computing and Information Sciences (CCIS)
Caraga State University
Ampayon, Butuan City, Philippines, 8600
eomorales@carsu.edu.ph

***Abstract* - Small to large companies handle multiple forms of records every day. These organizations could use these records for historical, demographical, sociological, medical, or scientific research and serve as benchmarks to measure the organization's future activities and decisions. The Department of Social Worker and Development (DSWD) Caraga continuously generates records daily. Still, their records management system is conventional, giving them a hard time retrieving and keeping track of the record's whereabouts. With this, DSWD Caraga embarks into record's digitization for its management to ensure the preservation of permanent and valuable papers, secured and accessible for future reference as required by the organization's different offices based on existing rules and regulations in records management. This paper endeavors to automate records classification using the open-source Python-Tesseract (PyTesseract) library, the wrapper for Google's Tesseract-OCR Engine. The process starts by converting paper-based documents into digital format (scanning) and then recognize and extract the text using the PyTesseract library. By integrating this library to Django and MySQL, management of record's classification, indexing, and archiving becomes easy. With the help of this system, record's safekeeping and retrieval bring comfort for the records officer.**

Keywords— PyTesseract, Digitization, OCR, Django, Records Management

I. Introduction

Small to large organizations deal with different types of records daily. According to the Department of Trade and Industry (DTI), there are 1.42 million registered businesses in the Philippines as of May 2019 [1]. These companies provide services or products to the customer. Thus, every documented transaction between these companies and customers will serve as their records that provide evidence that the activity took place. [2]An organization needs to keep accurate records to recall or prove what was done or decided in the past. Others may use these records for historical, demographical, sociological, medical, or scientific research. Traditionally, recording documents uses pen and ink or typewriter, but nowadays, computer program interaction is employed.

Records also serve as benchmarks to measure the organization's future activities and decisions[3]. These records are archived using the storage cabinet or shelves, which eventually lead to a high volume of documents. The retrieval of such becomes cumbersome, with data loss becoming unprecedented in natural calamities like fire and flood. A precise archiving methodology should be employed to address the proper record management. Improving storage, taking control, and supporting safe access are the components of appropriate archiving methods [5]. However, most companies cannot maintain them because of their limited workforce and human error.

The Department of Social Worker and Development (DSWD) Caraga, like any other organization, consists of several departments and continuously produces a high volume of records daily. These bulk of printed documents generated are piled and archived in the record's office and concerned divisions for safekeeping and future use. Presently, their records management system is conventional, giving them a hard time to retrieve and keep track of the record's whereabouts. Further, the supervision and manual indexing of documents generated each day burdens the records officer and is susceptible to human error. With this, DSWD embarks into record's digitization to ensure the preservation of permanent and valuable papers, secured and accessible for future reference in a more efficient way as required by the Department's different divisions.

Digitization is converting paper-based information into a digital format that includes computer-generated electronic documents or digital images produced by scanning or photography [6][17]. The physical documents stored in a cabinet, in a filer, or a folder are scanned and converted into digital form. This digitized data will be easier to preserve, access, and manage.

Previously, digitization of documents was achieved by manually typing the text on the computer. This method of converting paper-based documents into their digital form is time-consuming and vulnerable to human error. With the advent of high technology, various ways are emerging to digitize printed/paper-based materials. One of the most common and widely used methods is Optical Character Recognition (OCR)[23].

OCR involves detecting text content on scanned materials (like images) and translation of the recognized text to encoded text that the computer can easily understand. Through OCR, the digitization process is more comfortable as the document can be scanned, processed and the text extracted are stored in an editable form such as a word document [18].

978-0-7381-1177-3/20/$31.00 ©2020 IEEE

The process may not be 100% accurate and might need human intervention to correct some elements that were not scanned correctly. Error correction can also be achieved using a dictionary or even *Natural Language Processing* (NLP)[20]. A recent study asserted that the English language's precision level reaches up to 93% in the case of handwritten text and 98% for typed characters [21]. On the other hand, text extraction on Arabic text also gains more than 90% recognition and translation rate[22]. These results served as the basis for counting OCR as the best tool for transforming paper-based records into digital form

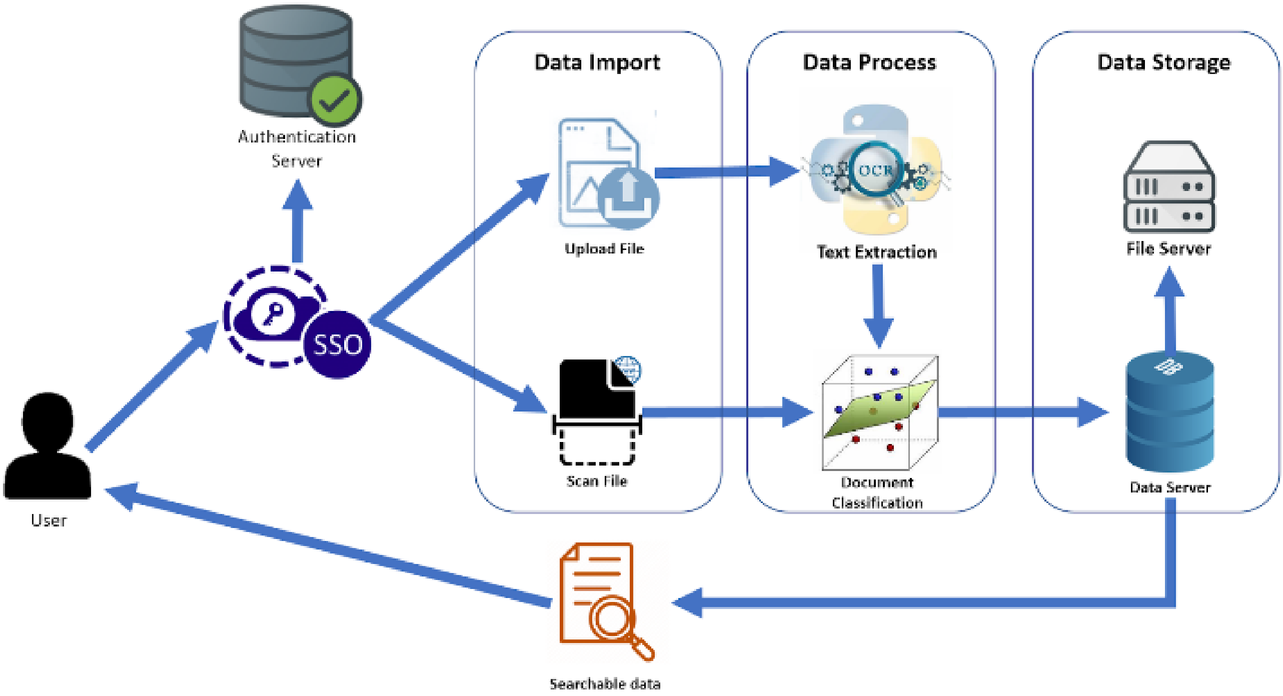


Figure 1 - System Framework

II. CONCEPTUAL BASIS

Transforming the vital and permanent records on paper into a digital form is the primary scope of this study. It focuses mainly on developing the Records Management System (RMS) that can cater to digitized documents, which shall help DSWD Caraga in converting paper-based records into digital format.

The application was conceptualized to enhance the conventional records management system of DSWD Caraga. Specifically, to assist the department in the following:

- Organized archiving and indexing of records that no longer require large physical space.
- Fast retrieval and tracking of records whereabouts not only for vital and permanent records.

For proper archiving and organized indexing, experts will assist in classifying the documents correctly. For fast retrieval of records, the application is equipped with a search facility where the result can be filtered or dropped down according to the user's specific criteria. On the other hand, to prevent data loss, an adequate setup of a secure data server with proper backup management should be in place.

Figure 1 shows the application concept, starting with the users undergoes an SSO authentication before the primary digitization process. Scanned documents will serve as the input to the system, then filtered for text extraction with N-grams-based linear classifier be applied for text recognition and perform the automated document class identification. After the AI-based classification, experts will review and modify the automatically-populated metadata fields to ensure accurate and complete data indexed and archived in the database and file server. With the search facility's help available in the application, finding and accessing the document is much more comfortable.

III. METHODS

A. *Data Preparation*

Prior to the development of the application, data were carefully identified and the system design underwent thorough analysis to fit into the holistic context of the department’s day-to-day activities.

Analysis and Design: The application was designed based on the Software Requirement Specifications (SRS) identified by the client (DSWD). The SRS lays out not only the functional and non-functional requirements, but it also includes a set of use cases that describe user interactions, access, and limitations. Database fields were also described exhaustively in the SRS to fit into the Department's data storage and archiving needs.

To understand better, maintain, and document information about the application, it utilizes the structural and behavioral UML (Unified Modeling Language) diagrams. Class, Object, and Activity diagrams were some of the system's visual representations applied.

The Class diagrams helps to define the attribute and behavior generated by the Object. And the object diagram illustrates the representation of every entity and its relationships involved in the system. Conversely, to describe the flow of different activities and actions from one to another, this was made clear through activity diagrams. Figure 2 below illustrates the process flow of the application.

Digital Imaging: To digitally archive the existing vital and permanent records on paper of the department, these records must be transformed into a digital image. Transforming paper-based documents into a digital form can be done through scanning or photography. With scanning, 300 dpi (dots per inch) were set, enough to have a realistic look from the original without compromising the resolution and file size. For efficient character recognition and text extraction PNG file format was used. All scanned documents will be uploaded in the DSWD Records Management System and will serve as the testing case for the records classification model of the system.

network. In order to monitor the status of uploading, the percentage of upload progress is indicated in the web interface.

In scanning, ScannerForWeb (SFW) was developed. It is a Windows Form Application that enables any web application to communicate to the local machine's attached scanner through a web socket protocol. It has a feature like text extraction. Mainly, the output of ScannerForWeb is divided into two parts: the data stream of the scanned document in BLOB or large Binary object, then the extracted text in string format. This output is sent through a "message" and can be received by the web application by establishing a WS (web socket) connection to ScannerForWeb by sending a "code" message, e.g., 1100, to invoke the form. This can be done through JavaScript, the front-end aspect of the web app. By default, SFW uses 8181 port ('ws://localhost:8181/').

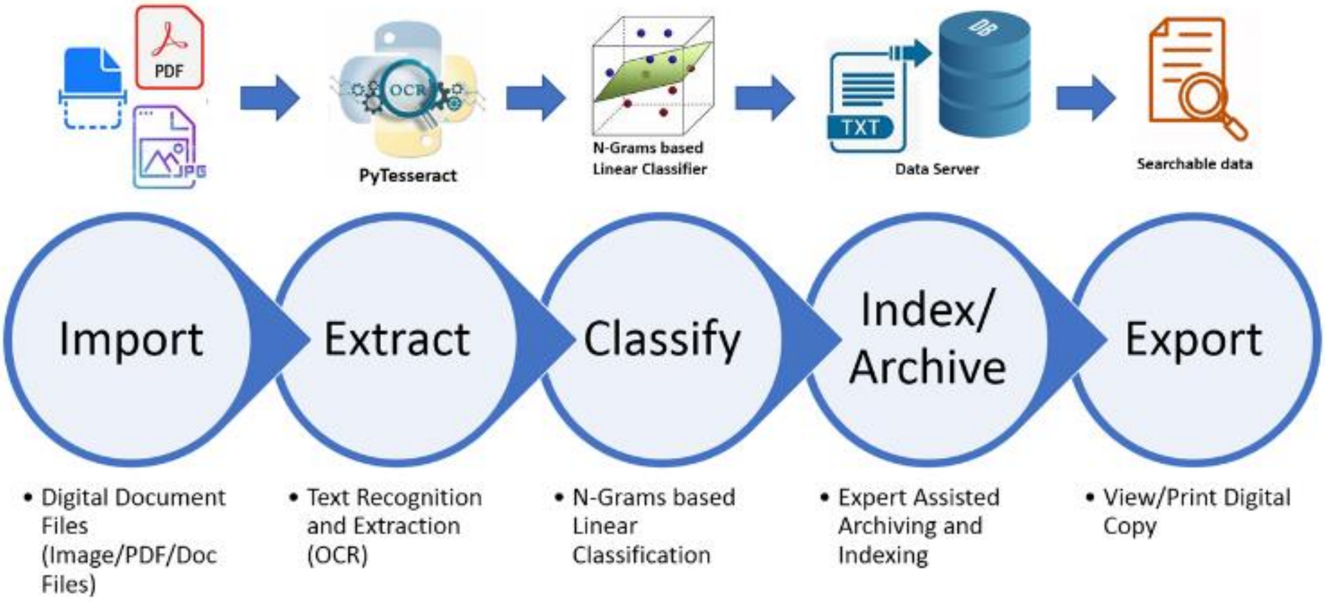

PDF
JPG
OCR
PyTesseract
N-Grams based Linear Classifier
TXT
DB
Data Server
Searchable data
Import
Extract
Classify
Index/ Archive
Export
• Digital Document Files (Image/PDF/Doc Files)
• Text Recognition and Extraction (OCR)
• N-Grams based Linear Classification
• Expert Assisted Archiving and Indexing
• View/Print Digital Copy

Figure 2- Digitization and archiving process flow

B. *Application Development*

Transforming the paper-based records into a digitally accessible format is the basis for the development of the application. It is derived from the five significant functionalities: Data Import, Text Recognition, Class Identification, Data Storage, and Data Retrieval. These functionalities became realized using the free and open source software.

i. Data Import

The storing and archiving process starts with data import. These data can be in the form of PDF, Image file, or document (.doc or .docx file types). For more efficient text recognition and extraction, png is the most preferred for image files. There are two ways to import files through "Uploading" and "Scanning."

The upload option is derived from the simple file upload mechanism through the web. With the integration of JavaScript libraries and Django templates and database plugins, storage and handling of uploaded files were made simple. In uploading files, only 20 files at a time were set to lessen the possible bottleneck of file transfer across the

SFW is written in C# using Microsoft Visual Studio 2019, an executable file installed only on a Windows OS machine. It runs in the background, with a minimalist design consisting of a dropdown selection for the available scanner, resolution, paper sizes, color modes or pixel type, and a start/stop scan button. SFW itself would not be useful unless integrated into another application such as a Document Classification System, wherein it can be utilized as a plugin for automated text extraction of scanned documents.

ScannerForWeb is an amalgamation of several existing technologies like NTwain (https://github.com/soukoku/ntwain) and tessnet2 (https://www.pixel-technology.com/freeware/tessnet2/). The NTwain provides an API for TWAIN access. It enables a particular application to access the TWAIN-compatible scanner resources. It is a .NET implementation of TWAIN, and it is open-source.

Some of the notable implementations of TWAIN is Dynamic Web TWAIN (https://www.dynamsoft.com/Products/WebTWAIN_Overview.aspx) – a browser-based document scanning SDK specifically designed for web applications. With just a few JavaScript code lines, one can develop robust applications to scan documents from all sorts of scanners, edit the scanned images and save them to a local/server file system or document repository. One of its editing features is it enables the user to rotate and crop images. It is also capable of text recognition (Optical Character Recognition) of PDF files.

On the other hand, for text recognition or extraction, IronOCR (https://ironsoftware.com/csharp/ocr/) is a good C# OCR library. It reads text and barcodes from scanned images and PDFs. Its output can be in plain text or JSON format.

Code Snippet 1. Sample code using IronOCR Library

```
using System;
using IronOcr;
//..
var Ocr = new AutoOcr();
var Result = Ocr.Read(@"C:\path\to\image.png");
Console.WriteLine(Result.Text);
```

The code snippet above demonstrates how simple to implement IronOCR; it just needs the image path then it outputs the result in text format. But this library is not open-source. For this reason, ScannerForWeb opts to use tessnet2 – a .NET 2.0 Open Source OCR assembly using the Tesseract engine. Tessnet2 is a .NET assembly that exposes straightforward methods to do OCR.Tessnet2 is multi-threaded. It uses the machine the same way Tesseract.exe does. Although Tessnet2's last update was in 2009, its accuracy is still good.

Given the availability and maturity of NTwain and Tessnet2 technology, these two are good options to develop ScannerForWeb – a Document Scanner and Text Extractor plugin.

ii. Text Recognition and Extraction

For text recognition and extraction, Python-tesseract (https://pypi.org/project/pytesseract/) is used, an optical character recognition (OCR) tool for python. That is, it will recognize and "read" the text embedded in images. Python-tesseract is a wrapper for Google's Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract. It can read all image types supported by the Pillow and Leptonica imaging libraries, including jpeg, png, gif, BMP, tiff, etc. Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file. Extracted text will automatically populate the metadata form.

Alongside the extraction of text is detection of document year, volume (number of pages), document author, setting up the digital location (file directory of the), record medium, physical location, and DRN (Document Resource Number). These fields will automatically fill out the metadata form.

iii. Record Class Identifications

Records were categorized into five major classes: administrative, financial, legal, personnel, and social services records. After extracting the text from the imported file, it is necessary to classify which category it belongs to for easy storage and document management. There are two methods used to categorize a record: first, the expert assisted classification. In this way, the expert identifies which class the imported documents belong to. Being an expert, he knows the details of the documents, including its metadata. The second is using the N-Grams based Linear Classifier Model. Extracted text will be fed to the model, and it will return an integer value that corresponds to the classification of the records. With the model, it can predict the class of the new record based on the classified documents. The classification accuracy using the model will depend on the number of records already classified and stored in the database. The more records organized, the more accurate the classification of the model is. At the early stage of the record's class using the model, expert assistance is still needed. The model needs to be updated as the record increases to ensure more accuracy.

iv. Archiving and Information Storage

The application has two (2) types of storage, the Database server where you can store textual information and the File server for storing imported files. MySQL Database, an open-source relational database management system capable of storing and managing vast volumes of data, will hold the records metadata and its related textual information. Utilizing the Django Model-view-template pattern, retrieval, and tracking of these records become faster. While the File Server is a dedicated server with ample storage to hold the department's current and future documents. The configuration of these servers requires internet connectivity.

v. Web Platform

The application uses Django, a python-based web application framework designed for fast and secure web application development. The integration of bootstrap libraries adds more aesthetics to the web pages. It is responsible for the system interfaces, modals, popups, and displaying textual information.

The information presented to the user was retrieved from the MySQL database. JavaScript is for webpage interactions, and some Django views interplay, like data sending and retrieval. JS could also be used for both the client-side and server-side, allowing the web pages to be interactive.

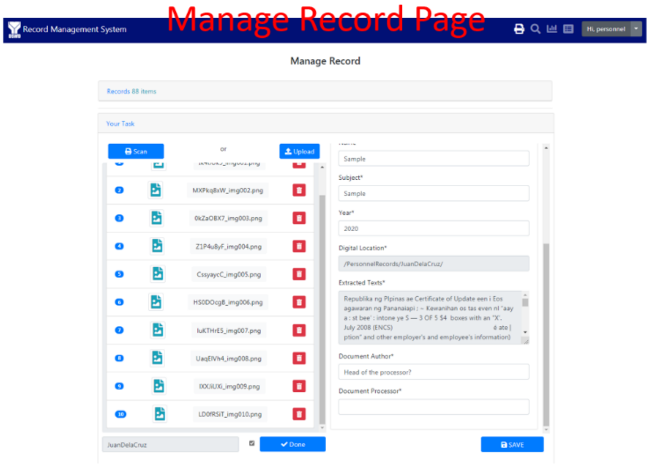
IV. RESULTS AND DISCUSSIONS

The Records Management System was developed to assist the DSWD records officer and its offices for fast record tracking that provide fast and efficient services. It is an application for digital archiving and indexing of documents cataloged into administrative, financial, legal, personnel records, and social services records. These records are vital in the organization and have a corresponding retention period where the documents could be disposed of as stipulated in the National Archives Policy of the Philippines.

The application development process starts by converting paper-based documents into digital format. It is done by scanning the bulk of records from the various divisions of the DSWD Caraga. The next step is the setup of a web application that can facilitate the uploading of the scanned documents and, at the same time, can automate the process of classification. It is done by utilizing Django – an open-source web application development framework. Django is then employed with PyTesseract – a Python OCR Library, to manage the recognition and extraction of text from the uploaded scanned files. Recognized text is then populated to the metadata form for record indexing. By integrating this library to Django and

MySQL, management of record's classification, indexing, and archiving becomes easy.

The application is equipped with a user-friendly GUI with four primary functional web pages: Manage, Search, Report, and Activity Logs Pages. Manage records page is the data entry page. On this page, data import happens like uploading and or scanning the documents, including the metadata form entry. Entries of the metadata form include the extracted text of the uploaded or scanned documents, classification, Document Resource Number (DRN), digital and physical location of the files, document author and file type, and retention period. While some of the entries of this form are automatically populated, the expert would review all the entries for a more accurate and correct file information tagging if necessary.



2	MXPkq8xW_img002.png
3	0kZaOBX7_img003.png
4	Z1P4u8yF_img004.png
5	CssyaycC_img005.png
6	HS0DOcgB_img006.png
7	luKTHrE5_img007.png
8	UaqElVh4_img008.png
9	lXXJiUXi_img009.png
10	LD0fRSiT_img010.png

JuanDelaCruz

Sample

Subject*
Sample

Year*
2020

Digital Location*
/PersonnelRecords/JuanDelaCruz/

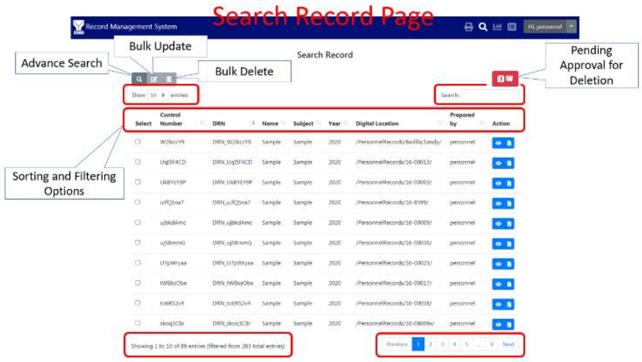
Extracted Texts*
Republika ng Plpinas ae Certificate of Update een i Eos agawaran ng Pananaiapi : ~ Kewanihan os tas even nl "aay a : st bee' : intone ye S — 3 OF 5 $4 boxes with an "X'. July 2008 (ENCS) é ato | ption" and other employer's and employee's information)

Document Author*
Head of the processor?

Document Processor*

Figure 3 - Application Manage records page

While data entry happens in the Manage records page, the Search Records page deals with the records entered. Searching, viewing, and data export is made possible through this page. The search record page is the landing page of the application. This page offers a search facility that has plenty of options for managing data in the archives. Managing the entries includes editing, updating and deleting of records. One feature of the application is the Watermark and Quick Response (QR) code affixed to the exported documents. It is in place for legal and security purposes.

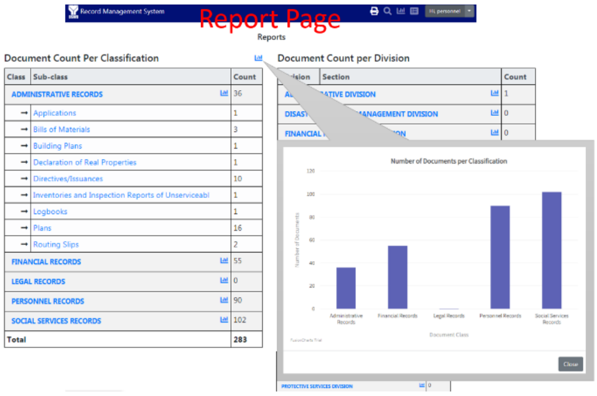


Select	Control Number	DRN	Name	Subject	Year	Digital Location	Prepared by	Action
□	W2lkccY9	DRN_W2lkccY9	Sample	Sample	2020	/PersonnelRecords/Badilla.Sandy/	personnel	
□	Uq|5F4CD	DRN_Uq|5F4CD	Sample	Sample	2020	/PersonnelRecords/16-00013/	personnel	
□	UkBYEY9P	DRN_UkBYEY9P	Sample	Sample	2020	/PersonnelRecords/16-00003/	personnel	
□	ufQSna7	DRN_ufQSna7	Sample	Sample	2020	/PersonnelRecords/16-8599/	personnel	
□	ujbkdAmc	DRN_ujbkdAmc	Sample	Sample	2020	/PersonnelRecords/16-00009/	personnel	
□	uj58mmG	DRN_uj58mmG	Sample	Sample	2020	/PersonnelRecords/16-00036/	personnel	
□	U7pWryaa	DRN_U7pWryaa	Sample	Sample	2020	/PersonnelRecords/16-00023/	personnel	
□	tWBksObe	DRN_tWBksObe	Sample	Sample	2020	/PersonnelRecords/16-00017/	personnel	
□	to6RS2v4	DRN_to6RS2v4	Sample	Sample	2020	/PersonnelRecords/16-00038/	personnel	
□	skoq3CBr	DRN_skoq3CBr	Sample	Sample	2020	/PersonnelRecords/16-00009x/	personnel	

Showing 1 to 10 of 89 entries (filtered from 283 total entries)

Figure 4 - Application search records page

The report page shows the statistics of the records in the database. Graphical charts represented records for a more comprehensive view. All the entries can also be printed and exported to several export formats (PDF, CSV, EXCEL, Print). Inventory of records can also be generated through this page.



Document Count Per Classification

Class	Sub-class	Count
ADMINISTRATIVE RECORDS		36
→	Applications	1
→	Bills of Materials	3
→	Building Plans	1
→	Declaration of Real Properties	1
→	Directives/Issuances	10
→	Inventories and Inspection Reports of Unserviceabl	1
→	Logbooks	1
→	Plans	16
→	Routing Slips	2
FINANCIAL RECORDS		55
LEGAL RECORDS		0
PERSONNEL RECORDS		90
SOCIAL SERVICES RECORDS		102
Total		283

Document Count per Division

ision	Section	Count
AI ATIVE DIVISION		1
DISAST ANAGEMENT DIVISION		0
FINANCIAL ON		0
PROTECTIVE SERVICES DIVISION		0

Figure 5 - Reports page of the records management system

The log page only shows the list of users' activity. These activities include the creation and modifications of entries. Delete and viewing of records can also be logged and displayed on this page for transparency of records.

Concluding Remarks

This paper presented an application that facilitates the digitization and management of paper-based records of DSWD Caraga Field Office. The application was devised to achieve organized storage and archiving of documents that no longer require large physical space. Purposely, this was developed to assist the records officer and the department for safekeeping and fast retrieval of records. Moreover, the system's capability to extract text from the uploaded image file makes the job more comfortable for the records officer to organize and store the records into the database. It becomes possible thru the integration of the PyTesseract library to the Django. Utilizing the records management system reduces, if not avoided, the possibility of data loss and damage due to natural calamity.

The application is presently hosted in the local server of DSWD Field Office Caraga, ICT Center. Before the server setup and configuration for production, the application underwent a series of user testing to polish its functionality..

Acknowledgment

This work is an output of the partnership project of Caraga State University (CSU) through the College of Computing and Information Sciences (CCIS) and the DSWD Regional Office Caraga. We acknowledge the DSWD Caraga Administration for entrusting CSU to develop its records management system. Grateful appreciation is also extended to the records officer and the DSWD Caraga staff for accommodating the projects' data collection team during the conduct of systems analysis and design.

References

[1] R. S. C. Canivel, "Number of registered firms in PH hits 1.42M," 21 May 2019. [Online]. Available: https://business.inquirer.net/270942/number-of-registered-firms-in-ph-hits-1-42m.

[2] Shepherd and G. Yeo, in Managing Records A handbook of Principles and Practice, London, Facey Publishing, 2003, p. XI.

[3] T. Kalusopa and P. Ngulube, "Record management practices in labour organizations in Botswana : original research," South African Journal of Information Management, vol. 14, no. 1, pp. 1-15, 2012.

[4] "About the Unilever Archive," Unilever, [Online]. Available: https://www.unilever.com.ph/about/who-we-are/our-history/unilever-archives.html.

[5] J. Jeston, J. Nelis and T. Davenport, in Business Process Management: Practical Guidelines to Successful Implementations Second Edition, Hungary, Elsevier Ltd., 2008, p. 141.

[6] Archive Principles and Practice: an introduction to archives for non-archivists, Crown, 2016.

[7] J. J. Johnson and J. J. R. McElroy, "Computer based records management system method". US Patent US5813009A, 1995.

[8] securedatamgt, "Secure Data Management," 23 March 2015. [Online]. Available: https://www.securedatamgt.com/blog/what-is-archiving/.

[9] Brian M. Mutale, Jackson Phiri, Web-based Document Archiving Using Time Stamp and Barcode Technologies-A Case of University of Zambia, Vol. 5, no. 4, p.4627, 2016.

[10] WCL Solution, "Digital Document Archiving and Management," WCL Solution, 2016. [Online]. Available: https://www.wclsolution.com/solutions/other-solutions/digital-document-archiving-and-management/.

[11] Smithsonian Institution, "Digital Preservation Challenges and Solutions," Smithsonian Institution Archives, [Online]. Available: https://siarchives.si.edu/what-we-do/digital-curation/digital-preservation-challenges-and-solutionshttps://siarchives.si.edu/what-we-do/digital-curation/digital-preservation-challenges-and-solutions.

[12] Simmi Dutta, Shruti, Haneet Kour, Chandani Bhagatt, Digital Document Archiving System with Optical Character Recognition. Vol. 1, no. 2, p. 41-42, 2013.

[13] G.S Lehal and ChandanSingh, Punjabi University, Patiala, Punjab, India, Feature Extraction and classification of OCR for Gurmukhi script. Vol. 12, p. 1-2, 1999.

[14] INTERNATIONAL JOURNAL OF ACADEMIC RESEARCH IN BUSINESS AND SOCIAL SCIENCES http://hrmars.com/hrmars_papers/Digitization_of_Records_and_Archives_Issues_and_Concerns.pdf

[15] OCR as a Service: An Experimental Evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym

[16] https://www.researchgate.net/publication/310645810_OCR_as_a_Service_An_Experimental_Evaluation_of_Google_Docs_OCR_Tesseract_ABBYY_FineReader_and_Transym

[17] J. Jayoma. A. Demetillo, M. Japitana, “Development of Land Property registry system application and dynamic database management for web-based GIS mapping: A case of Butuan City, Philippines”

[18] R. Smith, "An Overview of the Tesseract OCR Engine," Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Parana, 2007, pp. 629-633, doi: 10.1109/ICDAR.2007.4376991.

[19] Li L., Gao F., Bu J., Wang Y., Yu Z., Zheng Q. (2020) An End-to-End OCR Text Re-organization Sequence Learning for Rich-Text Detail Image Comprehension. In: Vedaldi A., Bischof H., Brox T., Frahm JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science, vol 12370. Springer, Cham. https://doi.org/10.1007/978-3-030-58595-2_6

[20] Graef R., Morsy M.M.N. (2019) A Novel Hybrid Optical Character Recognition Approach for Digitizing Text in Forms. In: Tulu B., Djamasbi S., Leroy G. (eds) Extending the Boundaries of Design Science Theory and Practice. DESRIST 2019. Lecture Notes in Computer Science, vol 11491. Springer, Cham. https://doi.org/10.1007/978-3-030-19504-5_14

[21] Banerjee S., Singh S.K., Rajib Bag A., (2020) Recognition of English Handwriting and Typed from Images using Tesseract on Android Platform. In: International Journal of Advanced Science and Technology Vol. 29, No.4, (2020), pp. 4042 – 4054

[22] Abdul Khader Jilani Saudagar & HabeebVulla Mohammad (2018) Augmented reality mobile application for arabic text extraction, recognition and translation, Journal of Statistics and Management Systems, 21:4, 617-629, DOI: 10.1080/09720510.2018.1466968