

Software Requirements Specification

for

Box.com/eDefender Integration

Version 1.0 approved

Prepared by James Yokley, Joseph Comeaux, Ethan Ngo, Alexander Voisan, Jimmy Castillo, Alexis Ponce, Karen Quan, Rosa Robles, Mario Avila, Sherina Marquez

Santa Barbara Public Defender's Office

09 December 2022

Table of Contents

Table of Contents.....	pg 2
Revision History.....	pg 3
1. Introduction.....	pg 4
1.1. Purpose.....	pg 4
1.2. Intended Audience and Reading Suggestions.....	pg 4
1.3. Product Scope.....	pg 4
1.4. Definitions, Acronyms, and Abbreviations	pg 4
1.5. References.....	pg 5
2. Overall Description.....	pg 6
2.1. System Analysis	pg 6
2.2. Product Perspective.....	pg 6
2.3. Product Functions.....	pg 6
2.4. User Classes and Characteristics.....	pg 7
2.5. Operating Environment.....	pg 7
2.6. Design and Implementation Constraints.....	pg 7
2.7. User Documentation.....	pg 8
2.8. Assumptions and Dependencies.....	pg 8
2.9. Apportioning of Requirements.....	pg 8
3. External Interface Requirements.....	pg 9
3.1. User Interfaces.....	pg 9
3.2. Hardware Interfaces.....	pg 10
3.3. Software Interfaces.....	pg 10
3.4. Communications Interfaces.....	pg 11
4. Requirements Specification.....	pg 12
4.1. Functional Requirements.....	pg 12
4.2. External Interface Requirements.....	pg 14
4.3. Logical Database Requirements.....	pg 14
4.4. Design Constraints.....	pg 15
5. Other Nonfunctional Requirements.....	pg 16
5.1. Performance Requirements.....	pg 16
5.2. Safety Requirements.....	pg 16
5.3. Security Requirements.....	pg 16
5.4. Software Quality Attributes.....	pg 16
5.5. Business Rules.....	pg 16
6. Legal and Ethical Considerations.....	pg 17
Appendix A: Glossary.....	pg 18
Appendix B: Analysis Models.....	pg 19
Appendix C: To Be Determined List.....	pg 21

Revision History

Name	Date	Reason For Changes	Version
First Draft	11/04/22	First Draft	1.0
Second Draft	12/09/22	Second Draft	1.0

1. Introduction

1.1 Purpose

This document is intended to give an overview on what we will use to achieve our goals for the Santa Barbara Public Defender's Office Box.com/eDefender integration project. The goal of this product is to utilize AWS Lambda with Microsoft Cognitive skills to automate certain processes within the Public Defender's Office and ultimately increase efficiency. This document will cover relevant software requirements for completion of this product..

1.2 Intended Audience and Reading Suggestions

This document is intended for developer operations, product developers, and users who are interested in an overview of the services and requirements used for the creation of this product.

1.3 Product Scope

Upon integration, the product, Visual Analysis Using Cloud Services, is aiming to speed up the Public Defenders ability to review cases involving digital audio or video evidence used in the evidence discovery phase.

This integration will configure specified Box.com folders to invoke an AWS Lambda function upon file upload. The files will be scanned by Microsoft Azure Video Indexer and utilize the machine-learning transcription, facial recognition, and key word recognition to assign timestamp tags and relevant metadata throughout the video or audio. Microsoft Azure Video Indexer service uses advanced machine-learning to detect people, items, themes, and language.

1.4 Definitions, Acronyms, and Abbreviations

- SRS: Software Requirement Specification.
- AI/ML: Artificial Intelligence Markup Language.
- VAUCS: Visual Analysis Using Cloud Services.
- Box: A cloud storage service.
- Box Skills: A framework for bringing Cloud Services to files stored on Box.
- AWS: Amazon Web Services.
- AWS Lambda: A serverless computing service on AWS.
- Azure: Microsoft Azure Cognitive Services

- VI: Microsoft Azure Video Indexer
- HTTPS: Hypertext Transfer Protocol Secure

1.5 References

- Serverless Framework documentation -
<https://www.serverless.com/framework/docs/providers/aws/cli-reference/deploy>
- Box Skills documentation -
<http://developer.box.com/guides/applications/custom-skills/setup/>
- Box.com documentation -
<http://support.box.com/hc/en-us/articles/360043696394-Create-New-Files-And-Folders>
- AWS Lambda documentation -
<https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>
- Microsoft Azure Video Indexer documentation -
<https://vi.microsoft.com/en-us>

2. Overall Description

2.1 System Analysis

The main problem this product is meant to solve is the increase of digital files, mainly video and audio, that the Public Defender is having to deal with, leading to an exponential increase in evidence that is required to be looked at. The goal is to create an integration that allows VI to scan videos and audio in order to speed up the process so we will use Box Skills in tandem with VI and AWS. Since Box Skills makes use of third-party AI/ML to get information within files, it isn't a standalone product and requires the use of other API; the box skills node is a middleware that will also need to be hosted on a serverless function within AWS Lambda.

The final product should be similar to many transcribing software currently on the market. One example is Youtube, as they have a function that automatically transcribes and uses those transcripts as subtitles for their videos.

2.2 Product Perspective

This software product integrates a variety of services consisting of Box.com cloud storage system, Box Skills, Amazon Web Services Lambda, and Microsoft Azure Video Indexer. This product is similar to a product used by Los Angeles Public Defender's Office to automatically transcribe and tag discovery media uploaded to Box.com. While Santa Barbara Public Defender's Office is looking to utilize the same services as Los Angeles, the motivation for creating a separate product is to curate the software to the specific needs of the public defenders. This product will require the extraction of transcript metadata to create a separate transcript that displays line numbers along with the transcription and possible translation to English. Box.com is a cloud storage service that can be extended using Box Skills developer applications. Box Skills can invoke an AWS Lambda function that connects to Microsoft VI to provide machine-learning transcription and facial recognition.

2.3 Product Functions

2.3.1 Box.com/Box Skills Upload Event

2.3.1.1 Box Skills monitors file uploads to Box.com folders and creates events

2.3.1.2 Box Skills event invokes AWS Lambda function

2.3.2 AWS Lambda Function

2.3.2.1 Connects and authenticates to Microsoft Azure VI

2.3.2.2 Utilizes VI API to perform transcription and facial recognition on file

2.3.2.3 Utilizes Box.com API to create metadata cards using VI data

2.3.3 Microsoft Azure Video Indexer

2.3.3.1 Uses advanced machine-learning to perform transcription and image recognition

2.3.3.2 Generates transcription and translation of audio

2.3.3.3 Generates timestamps of keywords and recognized faces

2.4 User Classes and Characteristics

There are two main user classes for this product, including Public Defender's Office software/IT administrators and lawyers/legal office professionals. A lawyer or legal office professional would use this product by uploading a discovery media file into the Box folder and waiting for the transcription and metadata tagging to be generated. Software and IT administrators would configure and maintain the systems' integration and credential management of AWS, Box Skill integrations, and Microsoft services.

2.5 Operating Environment

2.5.1 This software partially resides on Amazon Web Services cloud computing infrastructure

2.5.2 This software requires access to Box.com

2.5.3 This software requires access to Microsoft Azure VI

2.5.4 This software is accessible by most web browsers

2.5.5 This software requires an internet connection

2.6 Design and Implementation Constraints

2.6.1 SBPD authorization required to access Box.com folders

2.6.2 Operating costs of AWS Lambda and Microsoft Azure VI

2.6.3 Accuracy of transcription and keyword search is subject to errors and may require manual review

2.6.4 Accuracy of facial recognition is subject to errors and may require manual review

2.7 User Documentation

A User Manual will be provided in the future that outlines the process of uploading the file and receiving desired metadata. A Developer Operations Manual will outline the creation process of required services, credentials, and how to implement these services using Box Skills.

2.8 Assumptions and Dependencies

2.8.1 Amazon Web Services Lambda required to connect Box.com to Microsoft Azure VI

2.8.2 Microsoft Azure Video Indexer required to perform machine-learning transcription and facial recognition

2.8.3 Box.com is required for file cloud storage

2.8.4 Box Skills is required to integrate Box.com with Amazon Web Services

2.9 Apportioning of Requirements

2.9.1 Integration with eDefender through metadata notifications

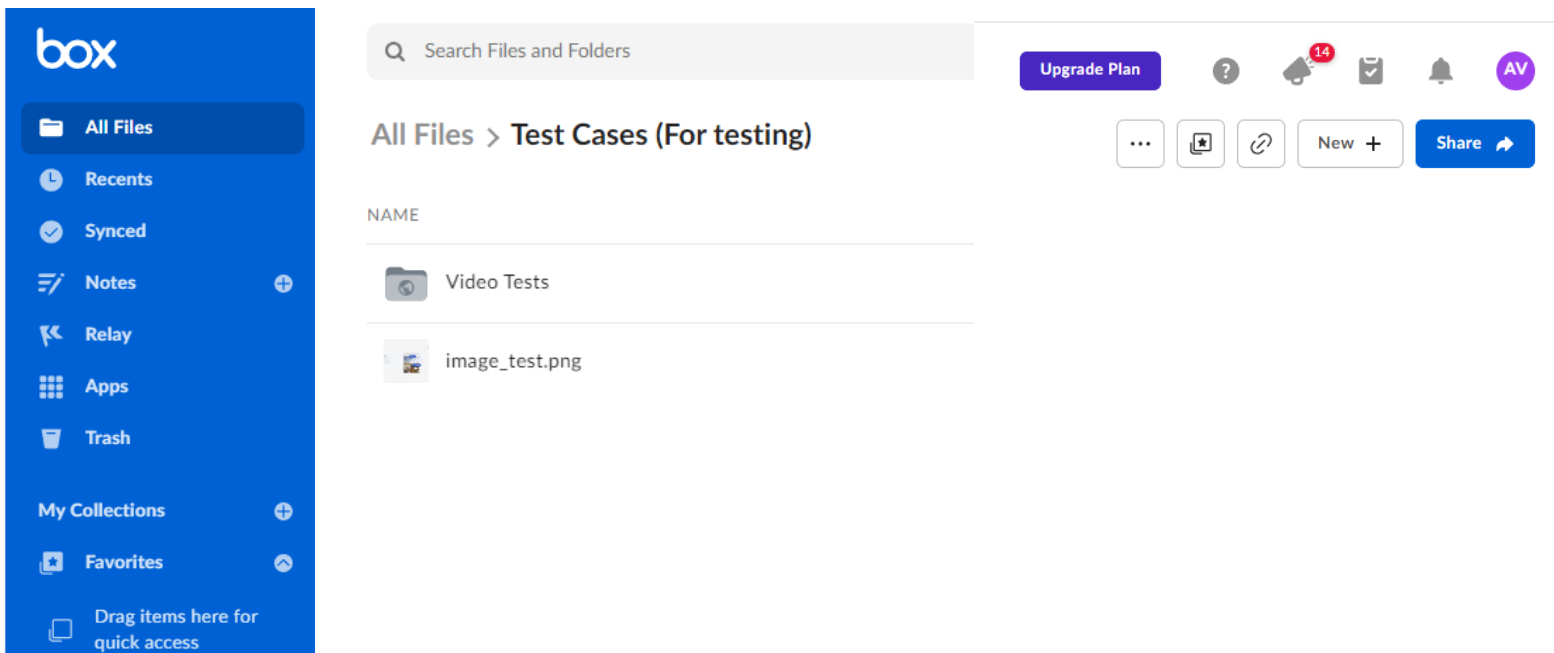
3. External Interface Requirements

This section of the document will provide a high-level overview of the various interfaces between the user and system.

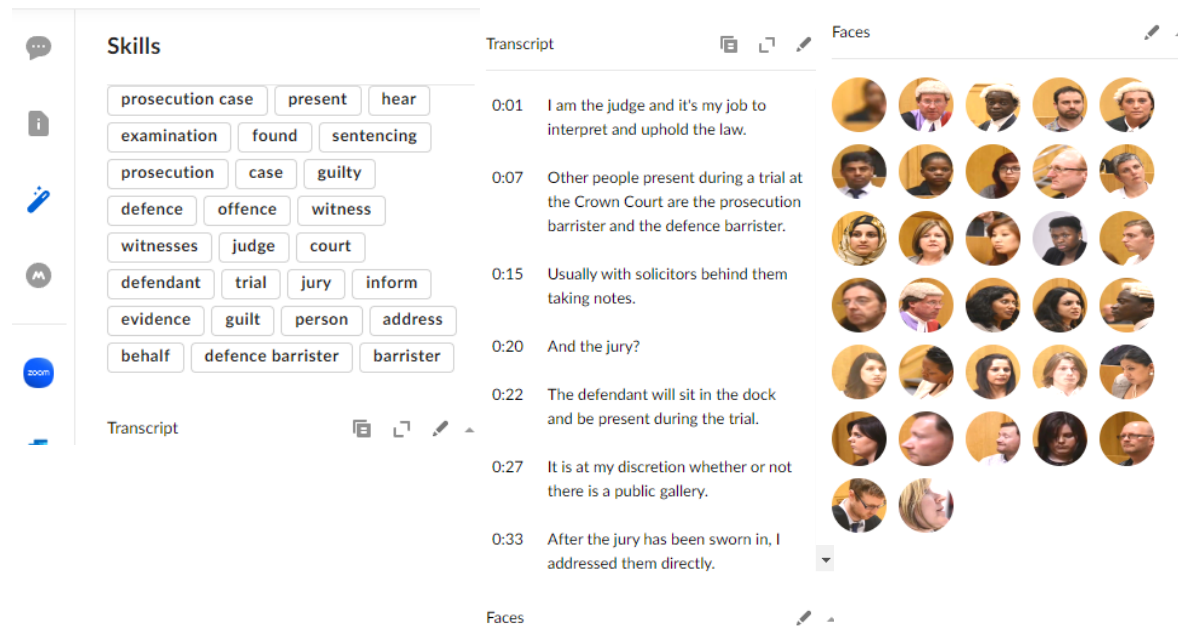
3.1 User Interfaces

A standard user will only be interacting with the product through the Box.com interface. The video indexing is performed on cloud services that do interact directly with the user. To upload, select a skill enabled folder, then press upload on the top right of the screen. Now just wait for completion and the file on Box will now display not only the content of the file, but its transcription amongst other things such as faces it recognized and key topics it comprehended.

3.1.1 Box.com interface to upload or search for files



3.1.2 Box.com file metadata displayed upon clicking on file



3.2 Hardware Interfaces

Hardware Interfaces N/A

3.3 Software Interfaces

3.3.1 Internet Browser

3.3.1.1 Microsoft Edge - latest versions (Chromium-based)

3.3.1.2 Safari - two latest released versions on macOS .

3.3.1.3 Firefox - The two latest, released versions.

3.3.1.4 Chrome - The two latest, released versions.

3.3.2 Operating System

3.3.2.1 Windows & Windows Server - The two latest released versions (64-bit)

3.3.2.2 macOS - The two latest major versions.

3.3.2.3 Android - All versions of Android released within the last three years.

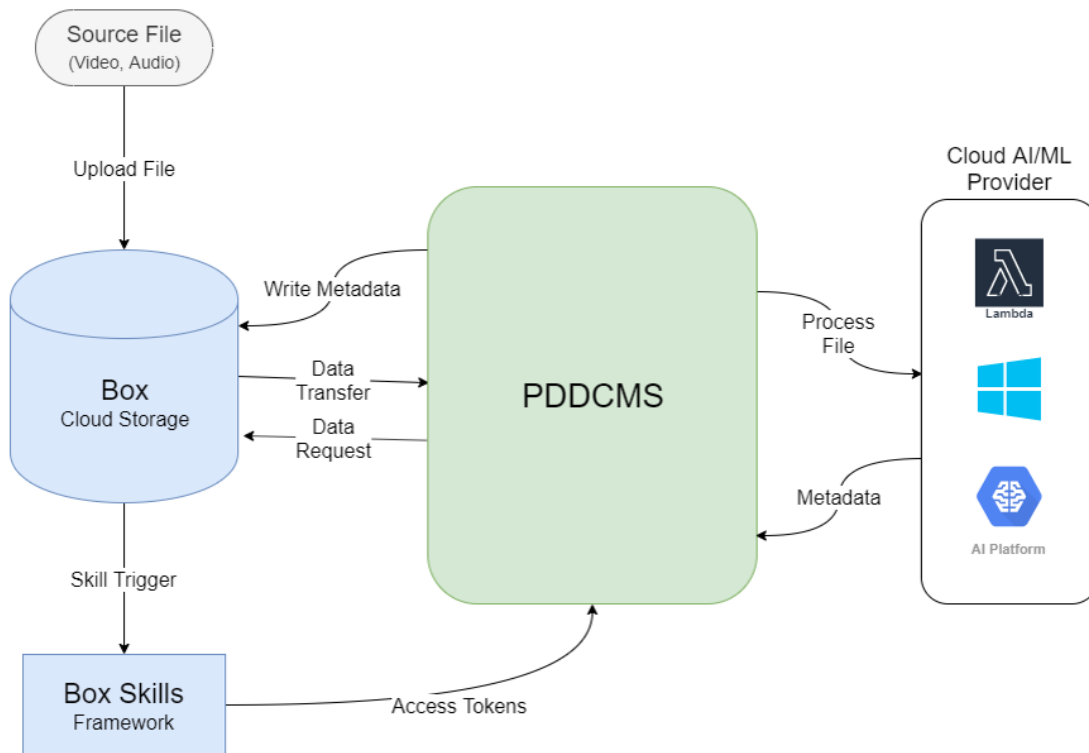
3.3.2.4 iOS - The two latest released versions.

3.4 Communications Interfaces

3.4.1 Hypertext Transfer Protocol Secure (HTTPS) GET

3.4.2 Administrators will be working with access tokens and an AWS server endpoint, including passwords needed for certain API Variables which are located in the user guide.

4. Requirements Specification



4.1 Functional Requirements

4.1.1 Box.com

4.1.1.1 The user shall have valid Box.com credentials with access to SBPD folders

4.1.1.2 The user should be able to view a file along with its metadata

4.1.1.2 The user shall have permissions to upload files to SBPD Box.com folders

4.1.1.3 Must be configured with a Box Skills Application

4.1.2 Box Skills Application

- 4.1.2.1 The user shall have administrator credentials to use Box Skills
- 4.1.2.2 The user shall have valid folder permissions to apply a Box Skill to a Box.com folder
- 4.1.2.3 Must create a custom skills application
- 4.1.2.4 Must be applied manually to each folder needing transcription capabilities
- 4.1.2.5 Must be configured with a valid AWS Lambda invocation URL
- 4.1.2.6 Must be configured to trigger event upon file upload
- 4.1.2.7 Shall provide access tokens to invocation URL

4.1.3 Microsoft Azure Video Indexer

- 4.1.3.1 The user shall have valid Microsoft Azure credentials
- 4.1.3.2 Credentials must be configured in AWS Lambda environment variables
- 4.1.3.3 Must account for uploaded media without audio before indexing
- 4.1.3.4 Must account for uploaded media without video before indexing
- 4.1.3.5 Must account for language translation before indexing
- 4.1.3.6 Must configure Azure skill for video indexing
- 4.1.3.7 Should periodically clear indexed files from Microsoft Azure

4.1.4 Amazon Web Services Lambda

- 4.1.4.1 The user shall have valid AWS credentials
- 4.1.4.2 Credentials must be configured in Serverless Framework environment
- 4.1.4.3 Lambda function must use Microsoft Azure VI credentials to authenticate
- 4.1.4.4 Lambda function must utilize Microsoft Azure VI API to access Microsoft machine-learning services
- 4.1.4.5 Lambda function must utilize BoxSDK API to access Box.com files and metadata cards

4.1.5 BoxSDK Developer

- 4.1.5.1 Must use BoxSDK API to manipulate Box.com files
- 4.1.5.2 Must use BoxSDK metadata tiles and cards to apply metadata to file
- 4.1.5.3 Must have valid Box Skills token to access Box.com files

4.2 External Interface Requirements

4.2.1 Box User Interface

- 4.2.1.1 Users will interact directly with the Box.com interface
- 4.2.1.2 Users can search files or folders by name
- 4.2.1.3 Users with valid permissions can upload files to Box.com folders
- 4.2.1.4 Users can interact with metadata displayed beside the selected file

4.2.2 Box Administrator Interface

- 4.2.2.1 Administrators will create Box Skill Application using Box Dev Console
- 4.2.2.2 Administrators will configure a Box folder with a Box Skill Application using the Box Admin Console
- 4.2.2.3 Administrators will configure a Box Skill using a valid AWS Lambda invocation URL

4.2.3 AWS Lambda Interface

- 4.2.3.1 Developers shall use the AWS Lambda developer console to create a lambda function
- 4.2.3.2 Developers shall use the Lambda console to modify and test code
- 4.2.3.3 Must use BoxSDK and Microsoft Azure APIs
- 4.2.3.4 Developers shall authenticate to Box and Microsoft Azure within the code
- 4.2.3.5 Developers must use the metadata generate by VI to fill Box.com metadata cards
- 4.2.3.6 Developers must use the access tokens to send the metadata cards back to Box.com

4.3 Logical Database Requirements

N/A. No Database storage is used with this product. All storage needs are satisfied through Box.com, AWS, and Microsoft Azure cloud services.

4.4 Design Constraints

- 4.4.1 All hardware is cloud based and maintained by either Box.com, AWS, or Microsoft Azure
- 4.4.2 Stable internet connection is required to upload or view files on Box.com
- 4.4.3 AWS Lambda outages will disable product functionality until service returns
- 4.4.4 Microsoft Azure outages will disable product functionality until service returns
- 4.4.5 Box.com outages will disable product functionality and content access until service returns
- 4.4.6 Computation cost for AWS Lambda and Microsoft Azure VI is subject to change
- 4.4.7 Microsoft Azure Video Indexer AI has varying effectiveness in non-English languages
- 4.4.8 Microsoft Azure Video Indexer AI has varying effectiveness depending on media quality

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- Analysis time of videos will take roughly 50% of the total length of the video.
- Accuracy of the AI model is dependent on clarity of video and audio. Users may see a reduction of accuracy up to 20% for audio transcription and 40% for visual tagging.

5.2 Safety Requirements

- This software integration is heavily dependent on AI and may be biased in ways unintended by the original creators of the AI.
- This software integration is heavily dependent on AI and may have certain inaccuracies when tagging people or items

5.3 Security Requirements

Security of external accounts of Microsoft Video Indexer and AWS accounts should be secured with strong passwords that ideally follow recommended safety standards.

5.4 Software Quality Attributes

Box Skill Application / API calls can be hosted in a number of ways. On a server, on serverless functions, or on a cloud server like AWS. The code is able to be migrated to other services, if the need arrives. However the code may need to be altered in order to facilitate such changes.

AI/ML services can also be replaced. This process is even simpler assuming we are still using AWS, as it only needs to direct AWS Lambda to make its call to a different service.

5.5 Business Rules

Users are highly restricted and should not be able to alter Box Skills, while admins should have full access to the code and other services used for this integration like AWS and VI.

6. Legal and Ethical Considerations

6.1 Privacy and security

6.1.1 Privacy and security are necessary aspects for SBPD to provide adequate legal representation to its clients

6.1.2 All laws regarding the handling of case discovery media must be followed

6.1.3 Access to case information should be limited to only those required

6.1.4 Protected health information and personal identifiable information should remain private

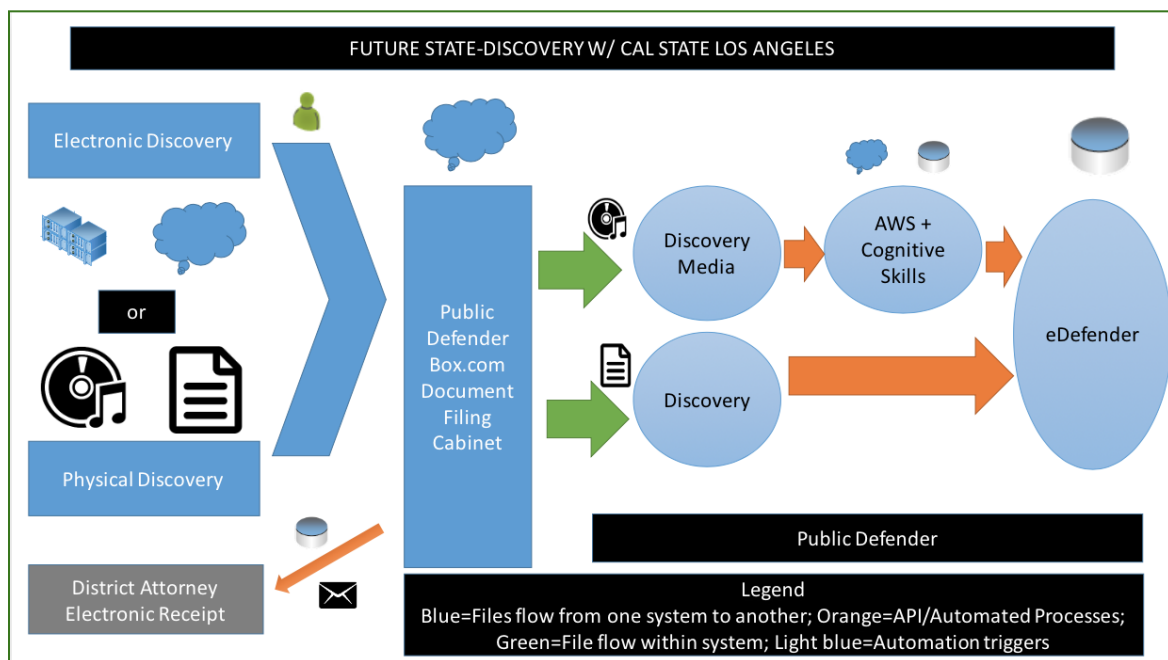
6.1.5 Credentials of all services must be tightly managed and two-factor authentication is encouraged

Appendix A: Glossary

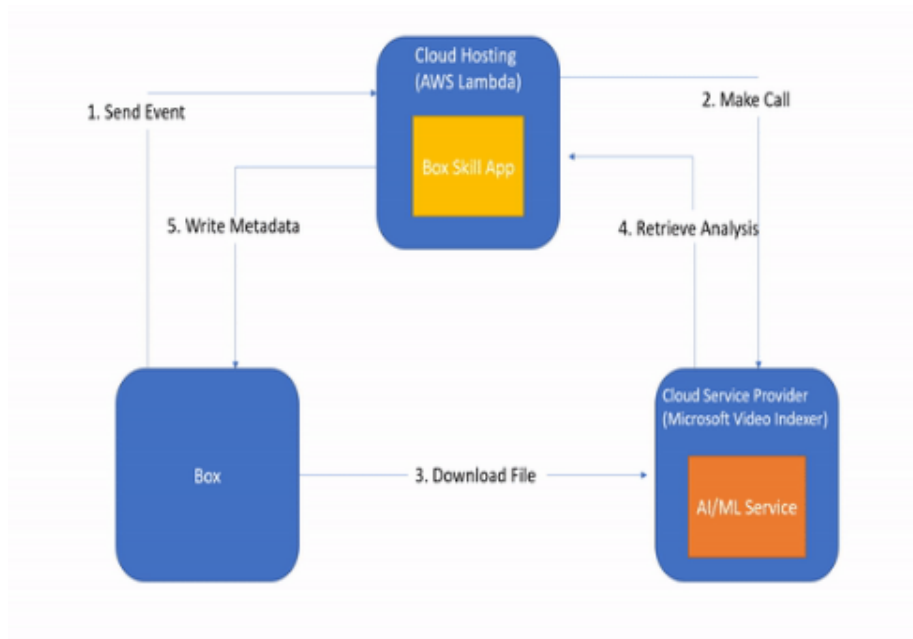
Reference section 1.4 to review all terms necessary to interpret this document.

Appendix B: Analysis Models

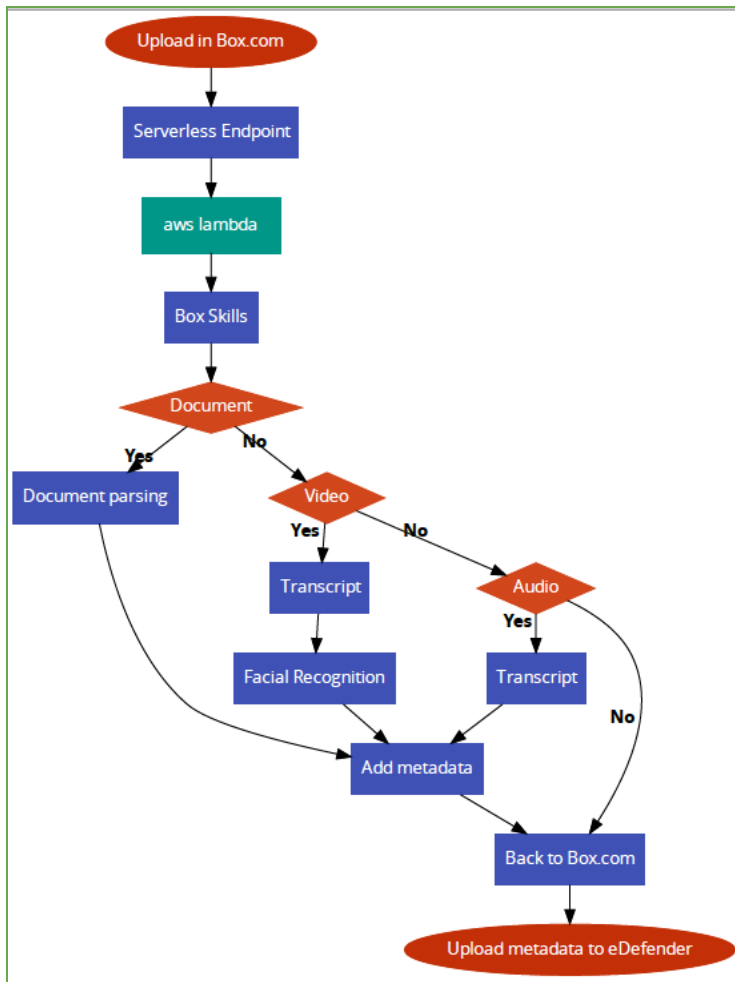
Discovery Process using Video Indexing:



Data Flow after Box.com file upload:



Alternate Data Flow image:



Appendix C: To Be Determined List

N/A