EZ DECRYPTER

Implementation Document

Michael Gabrys
Celina Vedro
Sara Cunningham
Yash Dube

Table of Contents

Executive Summary

Over the years, Cryptography and Cryptoanalysis have been important areas of focus for government agencies around the world in their efforts to solve crimes, thwart terrorism, and prevent illegal activities. Criminals are constantly creating new codes, symbols, and languages to communicate messages whilst avoiding/preventing government agencies from understanding them and intervening. One such example of this was the "Zodiac Killer" who took many victims in the Bay Area of central California in the early 1960's and 1970's. During this time, the killer had been writing encoded letters to the media that contained many different letters and symbols. Of course, at the time, there was no efficient means of decoding these messages and understanding what the killer was trying to say. Even now, some of these messages have not been deciphered due to the issue of there being no automated "system" solution that can handle the unique alphabet utilized by the killer. Another example of this relates to the United States' usage of the Navajo language during World War II to prevent the Germans from decrypting their communications. Using symbols and other things of that nature makes it difficult to digitally decode and encode messages. Decoding messages such as these via software solutions requires adding specialized or contrived alphabets using features such as optical character recognition and digitization. The problem is that currently there is not a single software solution that can do these things easily and efficiently in a one-stop-shop manner.

The following project has been designed to solve this problem and provide law enforcement agencies with a simple and effective way to decode and encode messages in an effort to aid in managing crime. Our team has created an application that integrates various commercial off the shelf technologies in order to provide one seamless interface for digitizing, encoding, and decoding messages. When a user takes a picture of the message, the application returns the decrypted message to the user in standard English or whatever their designated language may be.

The system developed by our team is broken down into three main components, digitization, translation, and decryption. First off, we utilize an Optical Character Recognition (OCR) software to capture an image of the message and digitize in such a way that it can be converted to plain text and then translated and decoded. Once the application receives this information from the OCR, it can then send it to the decryption/translation software. After these processes have been completed, the application receives the decoded/translated outputs and displays them to the user. The user is then able to see both the encoded and decoded messages at the same time and elect whether or not to continue with additional functions.

All software that will be integrated in our application has been selected with a focus on cost and comprehensive functionality. Adobe Scan and Cryptii are free to use and Google Translate is free up to 500,000 characters translated per month, which we do not envision any single agency crossing this threshold. Our focus on minimizing cost ensures that our solution to the problem is widely available to all law enforcement agencies.

To implement this application, agencies simply need to contact our customer service department who will provide a link to download our product. Users will also need to download the accompanying Adobe Scan from their respective mobile app store. Once this has been done and all accounts have been setup, the application is open for use to users upon authentication.

This system has been designed with simplicity, cost, and security in mind. Our team believes that the proposed application will more than adequately fulfill all deciphering and translation needs of

law enforcement agencies while providing them with the lowest possible cost for these services. If agencies have any questions, concerns, or requests, they may contact our customer service department who will then direct them to the appropriate parties.

Cost Analysis

The cost analysis table for the proposed solution can be found the following table. The entirety of the system will be hosted on the user's individual mobile device, with the various services hosted with their respective providers. All services to be integrated have been chosen on the bases of functionality and cost. The team has managed to keep the cost at $0, provided users do not exceed 500,000 characters of translation. Should additional needs or services be required, the cost analysis table may change accordingly.

Software	**Type**	**Cost**
Adobe Scan	OCR	Free
Google Translate	Translation	Free up to 500,000 characters. $20 per million additional characters (prorated).
Cryptii	Deciphering	Free
Total		$0

Documentation

The EZ Decrypter application is composed of five components: optical character recognition, language translation, deciphering, authentication, and the user interface. Depending on the task being completed, the required components communicate with one another in order to successfully complete the designated process. Adobe Scan OCR is hosted locally on the user's device along with the user interface. Cryptii, Google Translate, and authentication are hosted via the service providers as they are web-based services available over the internet. The local application communicates with web content via JavaScript Object Notation (JSON). The application sends translation requests to Google Translate and deciphering requests to Cryptii. Both services output a text string which is then returned to the EZ Decrypter application.

Authentication is be done via single sign on where the user is required to login to their respective government agency. Once an email address has been input into the application, a token is generated and transmitted it to the agency. If the user is currently logged in, access will be granted, and the user can begin use of the application. If the user is not logged in, the application will redirect the user to their designated agency website where they will input their credentials for validation. Upon successfully completing this process, the user will be granted access to the application. If the user is unable to be validated, the application will return an error message to the user. Should the user authentication be unsuccessful three times consecutively, the application will lockout the user for a designated period of time.

The user interface is composed of three main sections: login/authentication, image input, and deciphering. The interface includes various input controls to handle the functions required of the user. Text fields are used for user login credentials and deciphering text input/output. Image selection/capturing are done via buttons for capturing an image or selecting one from the gallery. Navigational buttons with icons are used to aid the user in navigating the application. Informational components such as message boxes and tooltips are used in order to output system messages to the user. All coding for the application is done with the JavaScript language and the ASP.NET framework via Xamarin.

Optical character recognition is done via Adobe Scan. The user of EZ Decrypter captures or selects an existing image which is then be shared with Adobe Scan via the use of an Intent class. This allows the applications to pass and receive results to and from one another. Adobe Scan receives the image file and then converts it to a PDF while also returning the plain text results to EZ Decrypter. The user is able to export the image to their local storage on their mobile device.

Translation and deciphering requests are done via the use of HTTP requests to either Cryptii, Google Translate, or both. A class has been created so that the processes may be handled accordingly. This class is utilized by the user interface once the use selects the desired function. It triggers the appropriate API for the given process. This allows the user to select multiple processes or rerun processes based on their needs. Requests are be managed via RESTful API which can handle multiple types of calls and return different types of data formats. This enables our use of universal JSON formatting, and the client and server can function independently. A layered system has been designed in order to handle the various actions and outputs.

The application utilizes the POST request method for transmitting data to Cryptii and Google Translate. This method enables the application to transmit requests to a web server where the data is enclosed in a request message. The data is then stored/submitted to the web application in the designated fields for translation/encryption. By using the POST method, the application can transmit various data types to web applications. Once the web application receives the data and it is transformed based on the request, EZ Decrypter then initiates a HTTP GET request to retrieve the output data from the web server which is returned to the application and displayed in the user interface.

User Training

As many government agencies and organizations will be utilizing our software, EZ Decrypter employees will help implement our service. Within 24 hours of purchasing our product, the EZ Decrypter team will contact you via email or phone to schedule a time for a virtual training session. The staff member will virtually walk you through all of the basics and answer any questions you may have. If there are any further questions or issues after the virtual training session, an EZ Decrypter member can provide on-site training if needed (only offered in select locations). Additionally, contact our customer service for any other questions/concerns.

Below is a user guide that covers the basics of our product. We recommend reading through the user guide before your training session for convenience.

Installation of EZ Decrypter

The application will be available for download via an embedded link contained in the email message sent to the user upon purchasing. Clicking the link will ask the user for permission to download and install the EZ Decrypter application on the user’s mobile device. The user will also be required to grant the application various permissions such as access to the user’s camera and/or gallery. Once these have been accepted and completed, the application will be available for use.

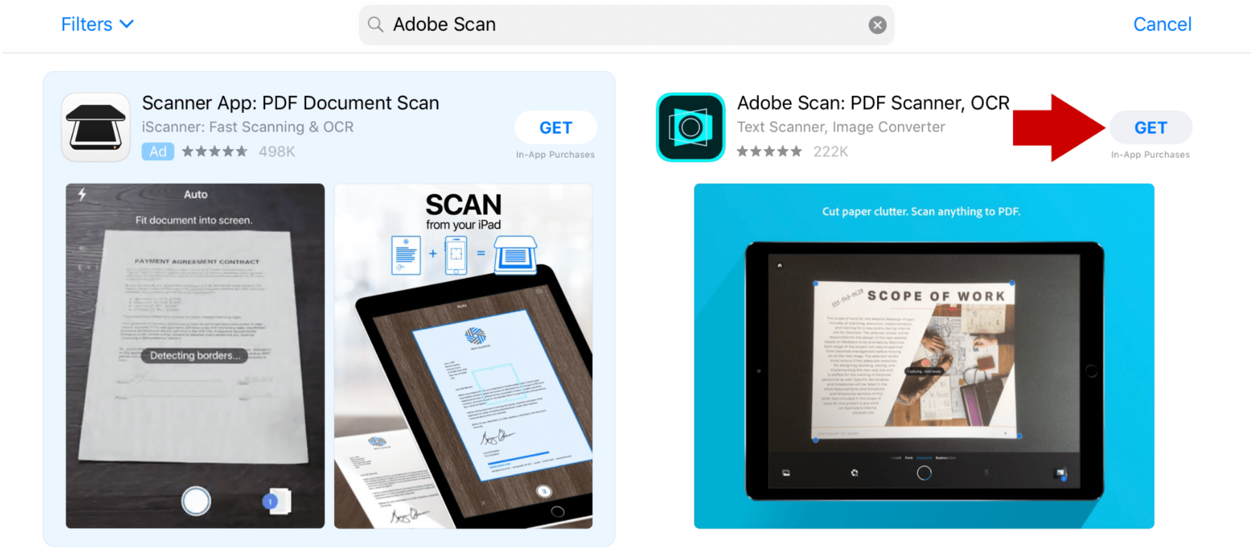
However, if a user begins to create a new decipher without the supporting Adobe Scan software installed, they will be presented with an error instructing them to download/update Adobe Scan.

Installation of Supporting Software

Adobe Scan

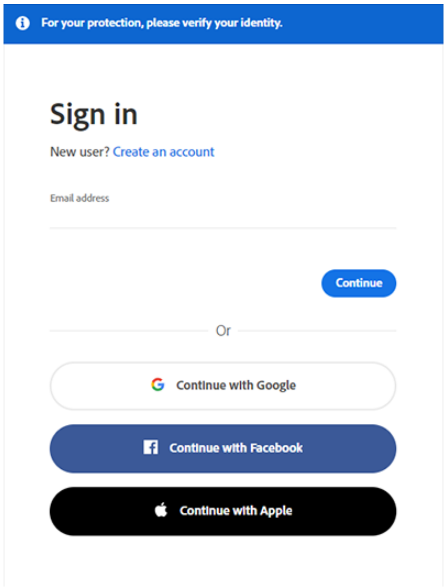
Adobe Scan is a free mobile app that scans documents into PDFs and automatically recognizes text. To download, go to the Google Play Store or the App Store and search "Adobe Scan." Once the search results come up, click on the first option (Figure 1). Next, click on the "get" button to download the app. You will be prompted to enter your account password to confirm your download.

Figure 1: Adobe Scan Installation


Filters
Adobe Scan
Cancel
Scanner App: PDF Document Scan
iScanner: Fast Scanning & OCR
Ad ★★★★★ 498K
GET
In-App Purchases
Auto
Fit document into screen.
Detecting borders...
SCAN
from your iPad
Adobe Scan: PDF Scanner, OCR
Text Scanner, Image Converter
★★★★★ 222K
GET
In-App Purchases
Cut paper clutter. Scan anything to PDF.
SCOPE OF WORK

After downloading the app, it should be on your mobile device. Find the app and open it. If it is your first-time using Adobe products, you will need to create an account. Follow the prompts on the screen to complete your account (Figure 2). If you already have an account, then login with your existing credentials. For additional help, please visit https://helpx.adobe.com/mobile-apps/help/adobe-scan-faq.html.

Figure 2: Adobe Scan Account Setup



For your protection, please verify your identity.

Sign in

New user? Create an account

Email address

Continue

Or

Continue with Google

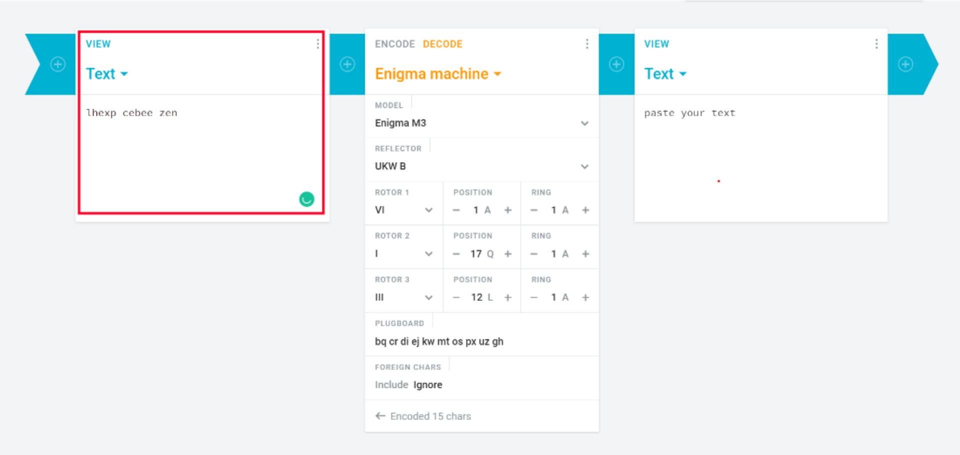
Continue with Facebook

Continue with Apple

Cryptii

Cryptii is a free web-based software which does not need any kind of installation. It can be directly accessed via the world wide web. The only requirement to use this is to have an internet access. EZ Decrypter will automatically communicate with Cryptii to obtain output results, so to the user this software is transparent and there is nothing they need to do in order for the application to function correctly. The web-based interface for Cryptii is depicted in Figure 3. To access the software, please visit: https://cryptii.com/

Figure 3: Cryptii Interface



VIEW

Text

```
lhexp cebee zen
```

ENCODE DECODE

Enigma machine

MODEL: Enigma M3

REFLECTOR: UKW B

ROTOR	POSITION	RING
ROTOR 1: VI	1 A	1 A
ROTOR 2: I	17 Q	1 A
ROTOR 3: III	12 L	1 A

PLUGBOARD: bq cr di ej kw mt os px uz gh

FOREIGN CHARS: Include Ignore

← Encoded 15 chars

VIEW

Text

```
paste your text
```

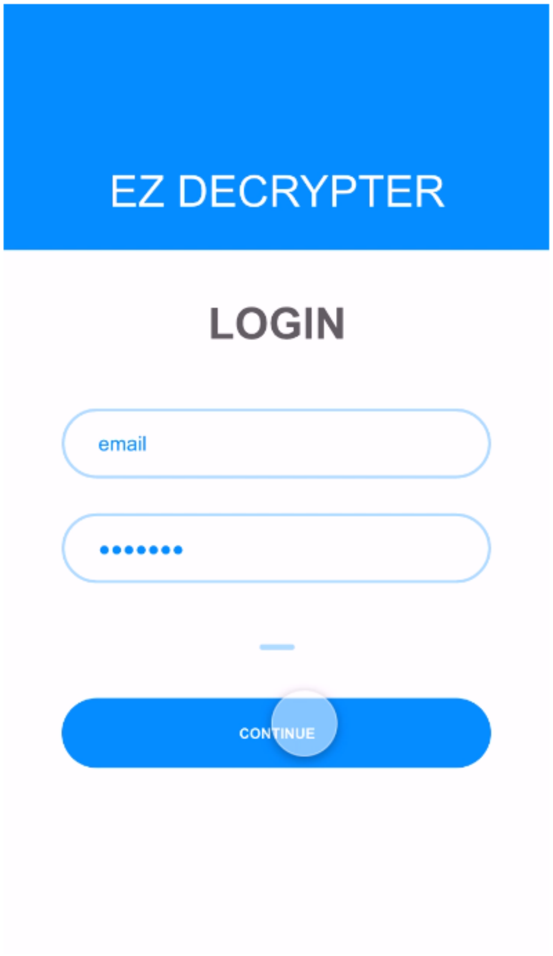
Google Translate

Google Translate API is another web-based solution for the application and the user does not need to do anything in order for it to function correctly. The only requirement is that the user has internet access so that requests can be communicated to and from Google Translate. This functionality will be transparent to the user.

Using EZ Decrypter

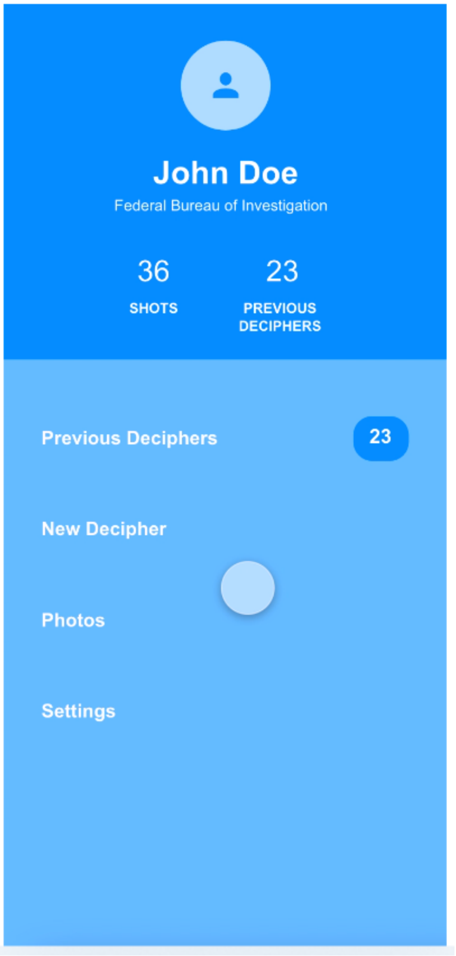
Upon opening the application, the user will be presented with the screen depicted in Figure 4. The user will be required to input their user credentials into the respective text fields and then click continue. Once this has been done, the application with validate the user and enable them to proceed with usage of the application.

Figure 4: Login Screen



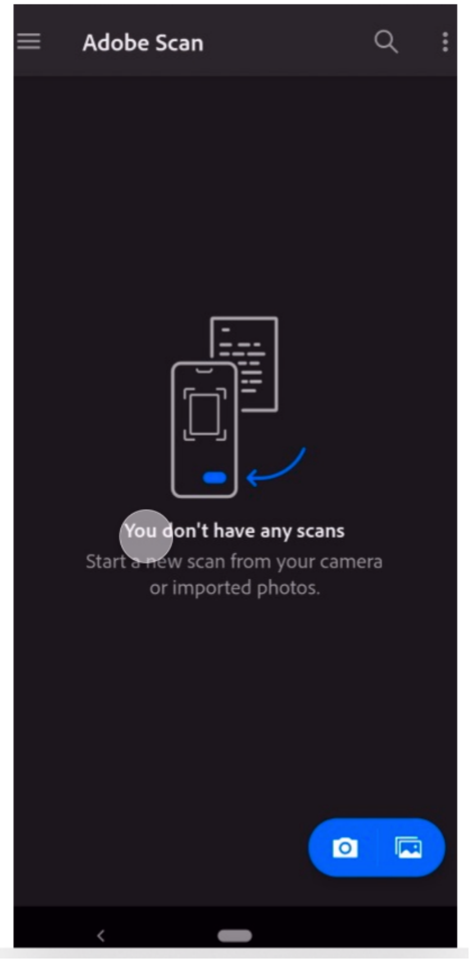
Next, the main home screen (shown in Figure 5) will be displayed to the user. From this screen, users can view previous deciphers, begin new deciphers, view photos that have been used or refer to the settings menu. The screen also displays various informational data such as username, picture, organization, image captures, and number of previous deciphers.

Figure 5: Home Screen



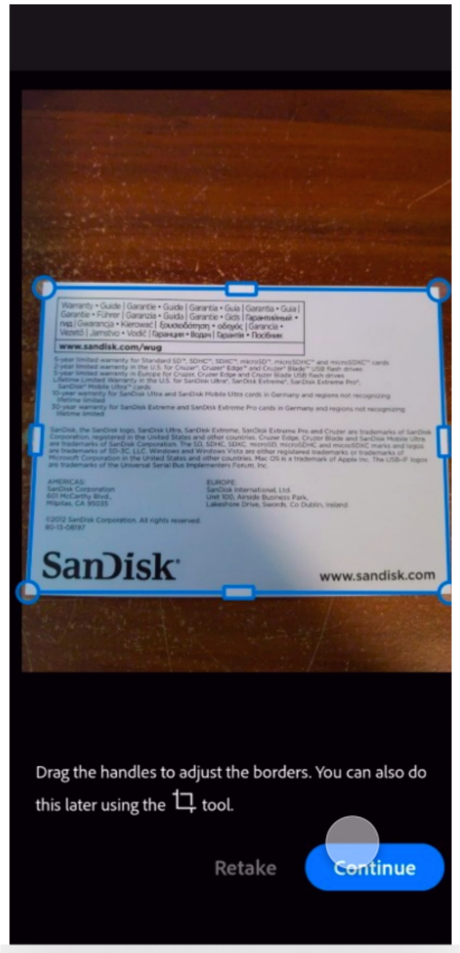
When a user clicks new decipher, they will be presented with the image capture/selection screen (shown in Figure 6). From this screen, a user may elect to take a new image or select one from their gallery. In addition to this, there are options to return to the main menu or search existing scans. If there have been previous existing scans saved on the device, they will be stored and displayed on this screen and the user will be able to select from them if desired.

Figure 6: Image Selection/Capture



If a new image capture is selected, the user will be presented with the selection screen shown in Figure 7. This screen allows the user to select what they wish to convert to text from what is captured by the camera. There is also an option to retake the photo if the user is unsatisfied with the current image. Once the desired areas have been selected, the user proceeds by clicking the continue button.

Figure 7: Image Capture Screen



Drag the handles to adjust the borders. You can also do this later using the tool.

The user is then taken to the image editing screen as shown in Figure 8. From this screen the user is able to adjust their image in various ways such as cropping, rotating, renaming, etc. This is also the screen that the user is directed to should they elect to use an existing image rather than capturing a new one. Once the user is satisfied with their image, they can save it as a PDF before continuing with the process.

Figure 8: Image Editing Screen

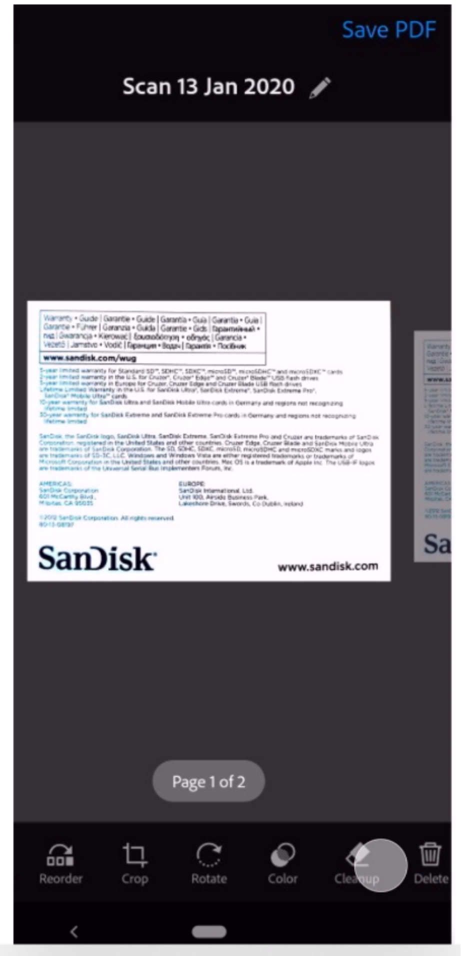
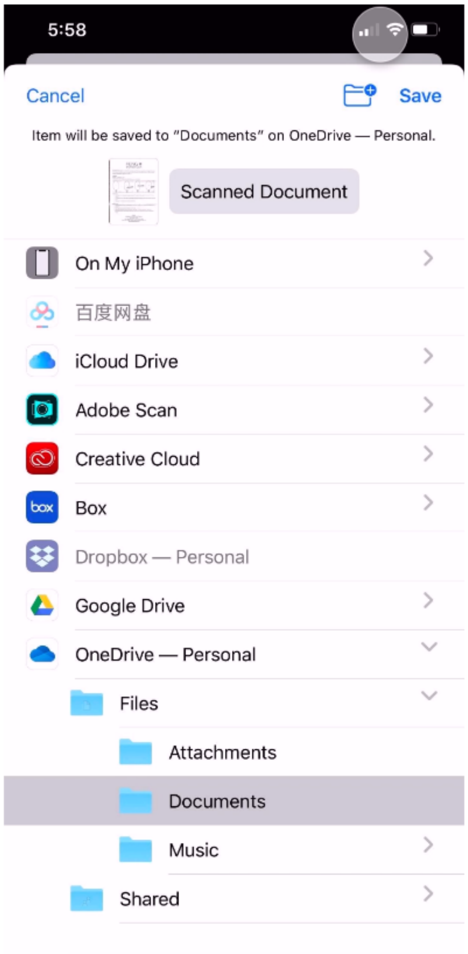


Figure 9 shows the exportation screen which allows the user to save the image in a variety of ways. This enables the user to store/share the image they have captured should the desire.

Figure 9: Exportation Screen

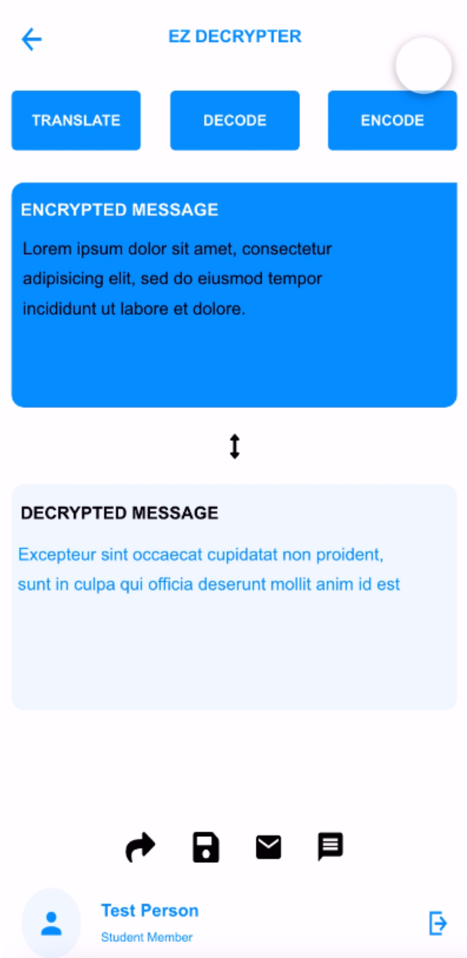


Item will be saved to "Documents" on OneDrive — Personal.

Scanned Document

The final screen presented to users in Figure 9 is the deciphering screen. From here, the beginning encrypted message that has been captured and digitized via adobe scan is displayed in the above text box. At the top, the user is presented with options to translate, decode, or encode the image. Upon choosing one of these options, the decrypted message field with automatically populate. Once the user has finished all deciphering needs, they may export the results, which will return them to the Exportation Screen. They may send the results via email to anyone they wish. At the bottom, information about the user is displayed alongside the icon to return to the home screen.

Figure 9: Deciphering Screen



ENCRYPTED MESSAGE

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore.

DECRYPTED MESSAGE

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est

This completes the end-user training of the EZ Decrypter application. If a user requires any additional training, they may contact our support team and resources will be provided accordingly.

Appendix A – Solutions Alternatives

Summary

This document discusses the various commercial off-the-shelf (COTS) and noncommercial off-the-shelf (non-COTS) solutions for decrypting and encrypting coded messages. The COTS software currently available each possess various pros and cons to solving the given problem. Many of them do not have the full functionality that would be required by the stakeholders for a comprehensive solution to their problem. Some applications could be used in conjunction with each other separately, or other NonCOTS will need to be created. To properly solve the project problem, it is required that a NonCOTS software is created while integrating COTS for best results. This strategy will create the ideal solution for the project and allow users to easily encrypt and decrypt messages without having to consult multiple platforms separately.

Optical Character Recognition

The first step required is to take a picture of the handwritten note so that it can then be digitized and converted to text. Nowadays, almost everyone possesses a mobile device with a camera, so this is what will be used to capture the image. Quality of images taken via mobile devices has drastically improved over the years so this will be an easy and efficient way to capture a high-quality image to be used. Once the image is captured, optical character recognition (OCR) technology can be used to convert any text or symbols in the image to plain text. This technology scans the image or document to automatically extract all printed or written text.

Commercial Off-the-Shelf Solutions

Listed below are the COTS solutions for optical character recognition currently available for use/purchase.

1. Adobe Scan - Free
    a. Top pick by The New York Times and provides a clean PDF with reliable text recognition
    b. Available on Google Play and Apple App Store
2. Microsoft Office Lens – Free
    a. A good choice if you are using Microsoft Office products and includes properly formatted OCR results
    b. Available on Google Play and Apple App Store
3. ScanPro – Free or additional subscription
    a. Includes extra features and sharing options
    b. More expensive on iOS devices
    c. Available on Google Play and Apple App Store

Non-COTS

The Non-COTS would be an application that is able to take a picture of the handwritten code and decode it all in one application rather than move from different systems. Having everything integrated in one application would be ideal, so there would not be any extra steps required. Though this is the perfect situation, there are some problems that arise. First off, everyone has distinctly different handwriting so the scanner and decoder would have to be extremely accurate. Second, there would have to be strong security within the application. If the images of the code and decryption were leaked, then that could cause some serious risks and issues.

Encryption/Decryption

COTS

There are many free online resources to decrypting files and texts. Most of which are only a google search away to access and begin decrypting.

1. Cryptii – Free
    a. Online tool to encrypt, decrypt, encode entered text and download it as a file

1) Codebeautify - Free
    a) Online tool for encrypting and decrypting text
2) Infoencrypt - Free
    a) Online tool for encrypting and decrypting text

Non-COTS

There are not many non-COTS that can easily be developed within a reasonable amount of time and within budget for this project. Due to this, we will be heavily relying on the available COTS and integrating them into our system. We will incorporate multiple tools in order to provide the most accurate and comprehensive results.

Translation Software

COTS

There are a variety of options when it comes to translating information. Most commonly, these COTS solutions are free to the public but there are a few subscription-based options available. For the process of this project, the team will be able to utilize some of the available websites to translate messages received. One of the most common translation services is Google Translate. It can recognize languages and translate to one of your choosing. There are issues with accuracy in meaning with "slang" translating from one language to another. Web services do not provide the same understanding as if a translator transcribed the messages. The information provided is still understandable to the user no less. These available sources make translating a handwritten encoded message easy from any language. The trouble arises when a "made up" language is used, making it more difficult to translate but still possible. The handwritten message, once digitized, can be processed through the website/software for translation. Translation software is going to be especially helpful with this system because it is necessary to understand the message before decrypting it.

1. Google Translate – Free
    a. Most widely known translation program
    b. Translates text, speech, and text with still or moving images
2. IBM Translate – Free
    a. Translates documents, apps, and webpages
3. OneHourtTranslation – Per Word Varying
    a. Human translation services
    b. $0.09 per word for general business content
    c. $0.14 per word for industry specific content

Non-COTS

Although there are translation services available, it is difficult to interpret handwritten messages that have been "made up". To overcome this difficulty, the team will create an application that can better decode the transcribed message. This process will include deciphering the unknown language into a usable format. Ideally, this would be more of a last resort considering the time that would be needed and how many services are currently available. If need be, the non-COTS solution would provide a better case-by-case basis for deciphering the "made up" languages.

Bespoke Primary Application

We are creating a bespoke solution, an ASP.NET-based web application, to integrate the disparate COTS solutions and supplement non-COTS solutions to fill the gap and deliver a full-fledged application. Our objective is to deliver a solution that is modular, customizable and offers agility for any organization to integrate it into their eco-system.

Front-End

The front-end application we found feasible is a Single Page Application (SPA) with ASP.NET Web API and Angular or REACT JS support. In our proposal, the entire page is loaded in the browser after the initial request, but subsequent interactions take place through Ajax requests. This means that the browser must update only the portion of the page that has changed; there is no need to reload the entire page. The SPA approach reduces the time taken by the application to respond to user actions, resulting in a more fluid experience.

Back-End

The back-end application is an essential part of our project as it will be connecting UI and different disparate technologies to bring them together through a common interface. We are planning to create a back-end application using ASP.NET Core MVC using Razor Pages version, a lightweight, open-source, highly testable presentation framework optimized for use with ASP.NET Core. In addition, it will also include a Web APIs framework for AJAX-based HTTP content-negotiation with built-in support to format data as JSON or XML.

Security

Since this is a web application, we are concerned about security. We are planning to implement a sign-in for our application using OWIN middleware (Microsoft, 2019). It is based on a traditional web browser-based solution using OpenID Connect. Our application will accept sign-ins of personal accounts from the likes of outlook.com and live.com. Additionally, work and school accounts from Penn State University and any company or organization that is integrated with Microsoft identity platform will be able to sign into our app.

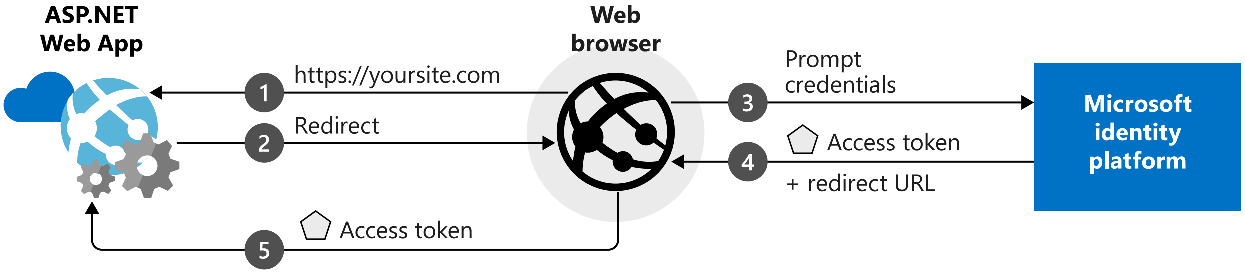


Figure 1 - OpenID Connect Authentication

We will be using following open-solution libraries.

Library	Description
Microsoft.Owin.Security.OpenIdConnect	Middleware that enables an application to use OpenId Connect for authentication
Microsoft.Owin.Security.Cookies	Middleware that enables an application to maintain a user session via cookies
Microsoft.Owin.Host.SystemWeb	Middleware that enables OWIN-based applications to run on Internet Information Services (IIS) by using the ASP.NET request pipeline

Storage and Hosting

We are looking for an affordable hosting solution. Currently we are planning to either choose a private cloud or preferable Microsoft Azure and utilize our free student subscription. Microsoft Azure is the preferred solution as Microsoft provides a Student subscription that gets us $100 monthly credit to spend, and later, if required, we can decide if we want to upgrade to pay-as-you-go pricing and remove the spending limit. Moreover, a cloud-based solution offers flexibility and allows us to collaborate as a group efficiently.

Conclusion

We are planning to use both COTS and nonCOTS solutions in our project due to the fact that current COTS by themselves do not adequately provide all the functions that we require for this project. Given the time and budget constraints, we will be relying mostly on the Commercial-off-the shelf solutions to enhance our system and provide basic functions. However, non-COTS software will need to be developed in order to completely solve the given problem and provide a fully functional system.

Appendix B – Design Requirements

These are the minimum requirements for the mobile device in order to be able to utilize the EZ Decrypter application:

- It should have minimum Android 6.0 as its operating system.
- It should have minimum 3 to 4 GB RAM.
- It should have at least a HD (12mp) camera to take clear pictures.
- The device should have approximately 600MB storage to run the applications.

Main System Modules

Overview

The main system modules include various software and hardware components that comprise the EZ Decrypter. The system enables the users, such as local law enforcement, the legal system, FBI, and NSA, to use our product to decrypt messages.

The application will be using the Xamarin Forms framework to develop the application so it can be run on iOS, Android, and Windows. By using the Xamarin framework, the application will be available to all users using any mobile platform. The application includes three components:

- Front-end web development enables users to use and interact with the system. The EZ Decrypter will be utilizing a Single Page Application (SPA) with Xamarin Web API and React JS. Using a SPA provides a better user experience, minimizes the number of pages to be designed and has a faster load time.
- Back-end development will connect the server that communicates between the database and the browser. The back-end development is just as important, if not more important than the front-end development. The application will be using ASP.NET Core MVC using Razor pages.
- The predictions for security threats and attacks are increasing each year exponentially, so the user's data security is critical. The application will include a single sign-on (SSO) using OWIN middleware. The SSO will only accept accounts associated with approved law enforcement agencies.

Authentication

Authentication will be done via Single Sign On in conjunction with law enforcement agency platforms. To utilize our application, users will be directed to their agency's login page. Once they have input their credentials and logged in, they will be returned to the application and permission will be granted. By requiring users to login this way, it will prevent any unauthorized users from utilizing the platform. In addition, usage of an SSO eliminates the need for development of a comprehensive native login system, maintenance, and storage of user credentials, etc. The application will generate certificates via Security Assertion Markup Language (SAML) for added security during authentication. This method creates a token that is transmitted between the law enforcement agency and our application to validate the user instead of transmitting user credentials which can make them vulnerable to being compromised.

Optical Character Recognition

For optical character recognition (OCR), our application will be using Adobe Scan. This is a mobile application that can scan text via camera and convert it to a readable PDF. Adobe Scan utilizes OCR technology alongside AI-Enhanced scans in order to clearly digitize image text without glares and shadows. Using this technology allows easy digitization of messages so that they can then be encoded, translated, and decoded via other software.

Language Translation

The system will utilize the Google Translate API in order to translate messages to/from languages based on the needs of the user and to also promote successful encryption and decryption. Google Translate charges based on the number of characters that has been transmitted with the first 500,000 characters per month being free. After that, the cost is prorated with $20 for every million characters.

Encryption/Decryption

Encryption and decryption will be done via Cryptii which is an open-source conversion, encoding, and encryption web application. It can encode, decode, encrypt, decrypt, convert, and translate any content that is input. It achieves this by utilizing various ciphers, formats, algorithms. Cryptii has been chosen as for its comprehensive methods and features as well as being free to use.

Inputs and Outputs

Overview

Our application is based on mobile platform so all input will be through our application in the mobile device. Adobe Scan, Google Translate, and Cryptii will be the main modules responsible for the inputs and outputs outside of the application. Adobe Scan will be responsible for scanning, digitizing, and converting the image files into plain text, whereas Cryptii will be decrypting the text and Google Translate will be translating text.

Input

Authentication

For authentication, the user will input their email address which will be collected via scanners embedded in the application's Java code. Once this has been done, the application will create a token via SAML and send it to the Identity Provider for authentication. Should the user be required to login, they will be redirected to the Identity Provider's login page which will handle any additional required inputs.

An example of Java code that utilizes a scanner is as follows:

```
import java.util.Scanner;
Scanner input = new Scanner(System.in);
String email = input.nextLine();
input.close();
```

Optical Character Recognition (Adobe Scan)

Adobe Scan can collect its own inputs by using the user's camera to take a picture, converting it to PDF and then to text. Adobe Scan will also accept images from the user's gallery in various formats such as .png and .jpg.

Language Detection (Google Translate)

Google Translate requires string inputs in order to provide translation. This string will be provided by EZ Decrypter based on the outputs from Adobe Scan and Cryptii.

Encryption/Decryption (Cryptii)

Cryptii requires string inputs in order to provide encryption and decryption. This string will be provided by EZ Decrypter based on the outputs from Adobe Scan and Google Translate.

Output

Authentication

Authentication will produce no outputs to the user unless their authentication requires login, in which case the application will direct them to their agency's login page. Should the email address provided to the application be invalid to all Identity Providers, the system will produce the login error "Invalid Email Address". This error message will be stored as a string in the database.

Optical Character Recognition (Adobe Scan)

Adobe Scan will output plain text in the form of a string to be used by the application.

Language Detection (Google Translate)

Google Translate will output plain text in the form of a string to be used by the application.

Encryption/Decryption (Cryptii)

Cryptii will output plain text in the form of a string to be used by the application.

Processes

Authentication

Users must utilize the single sign on via their designated law enforcement agency in order to begin using the application. Once the credentials have been verified, the application will validate the user and allow use of the application. If the credentials cannot be validated, it will direct the user to their organization's login page or present them with an error message. Should a user fail to successfully login in more than three times, the application will lock for a period of time. The login process is depicted in Figure 1.

Figure 1: Login/Authentication

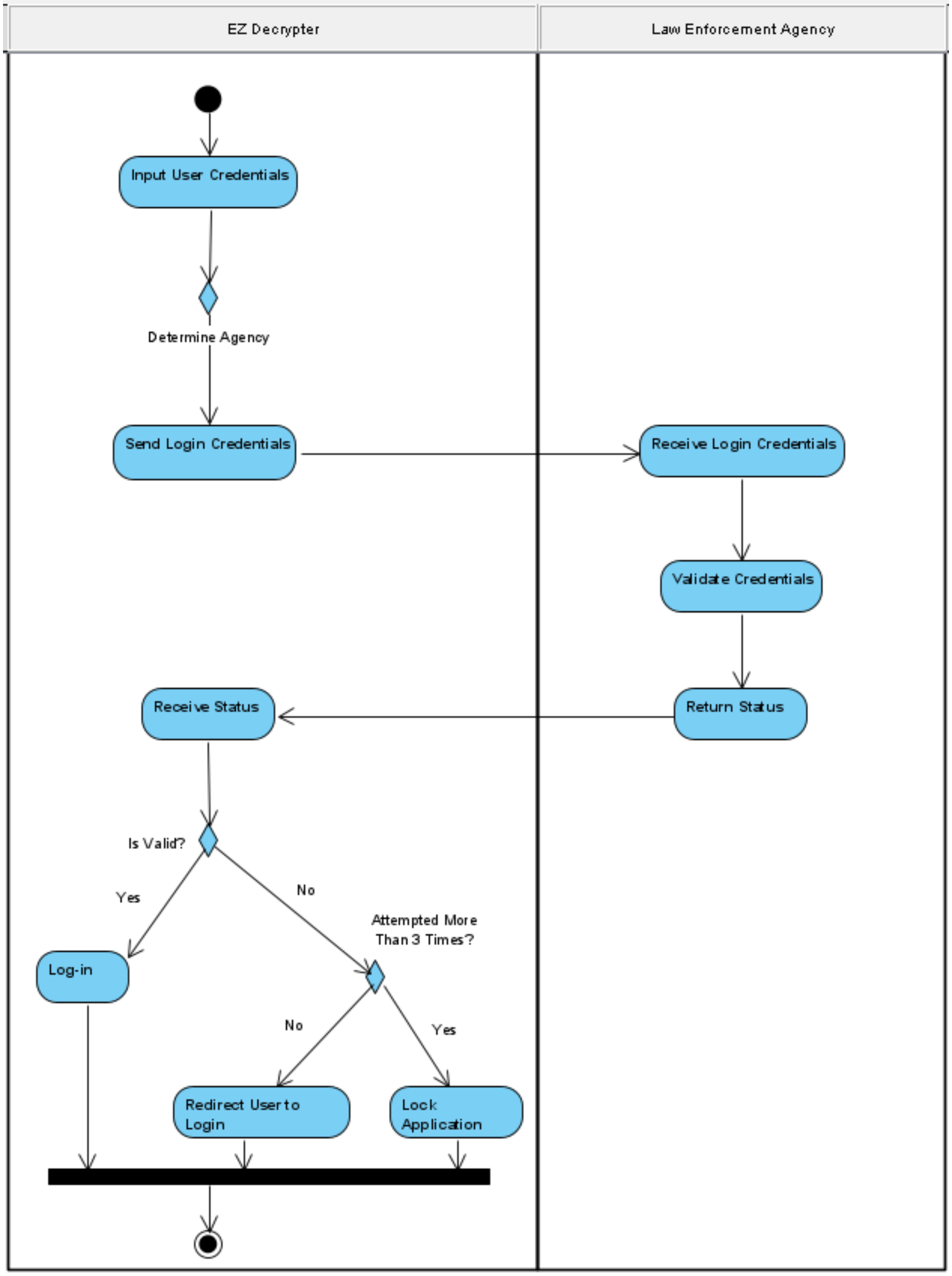
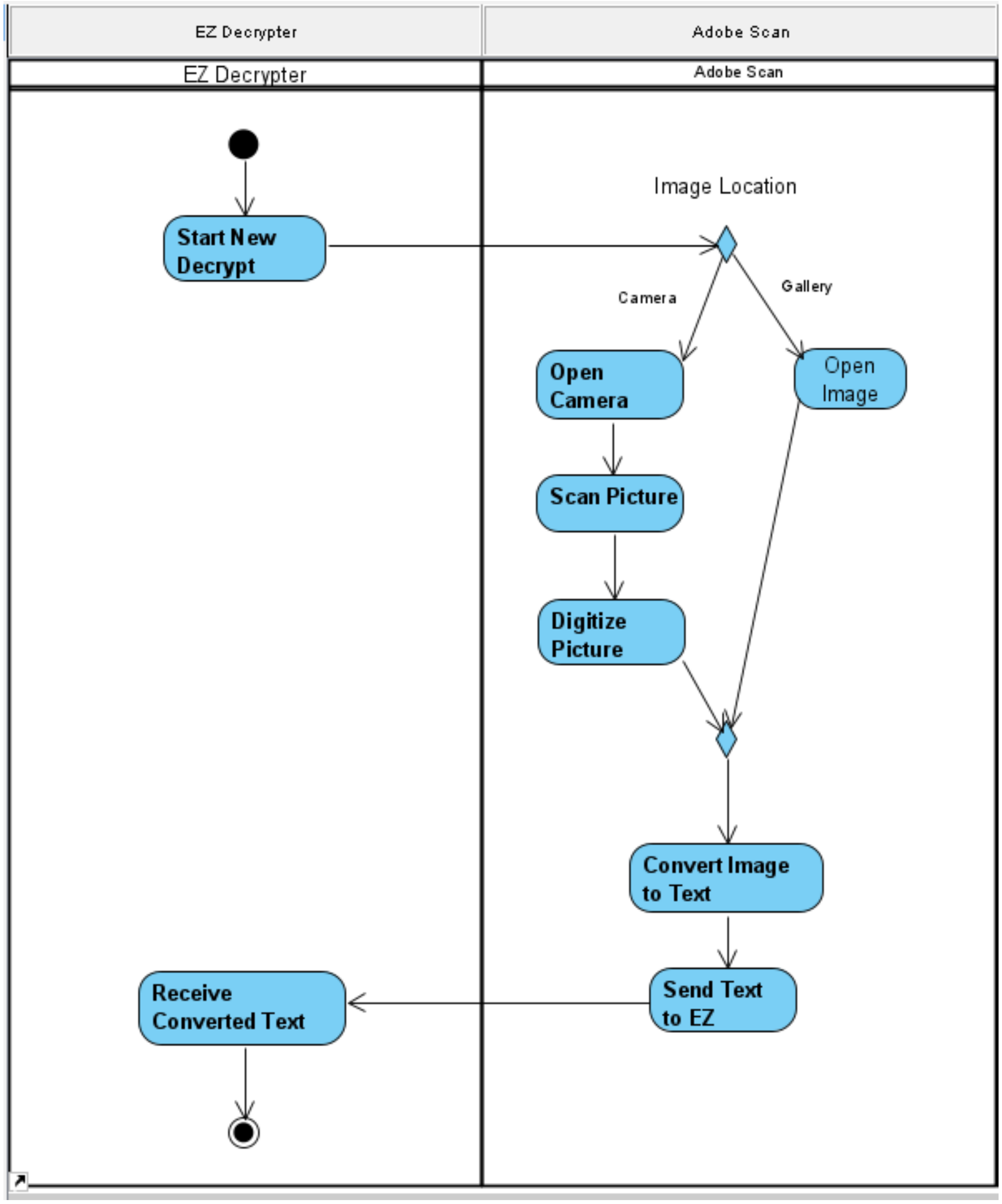


Image Conversion

Once the user is validated, they begin by starting a new Decrypt. This initiates Adobe Scan to open which in turn opens the user's camera in order to capture the image of the text to be translated or it accepts an image stored in the user's gallery. Adobe Scan then takes the picture, digitizes it, converts it to a PDF and then to plain text which is received by EZ Decrypter. This process is depicting in Figure 2.

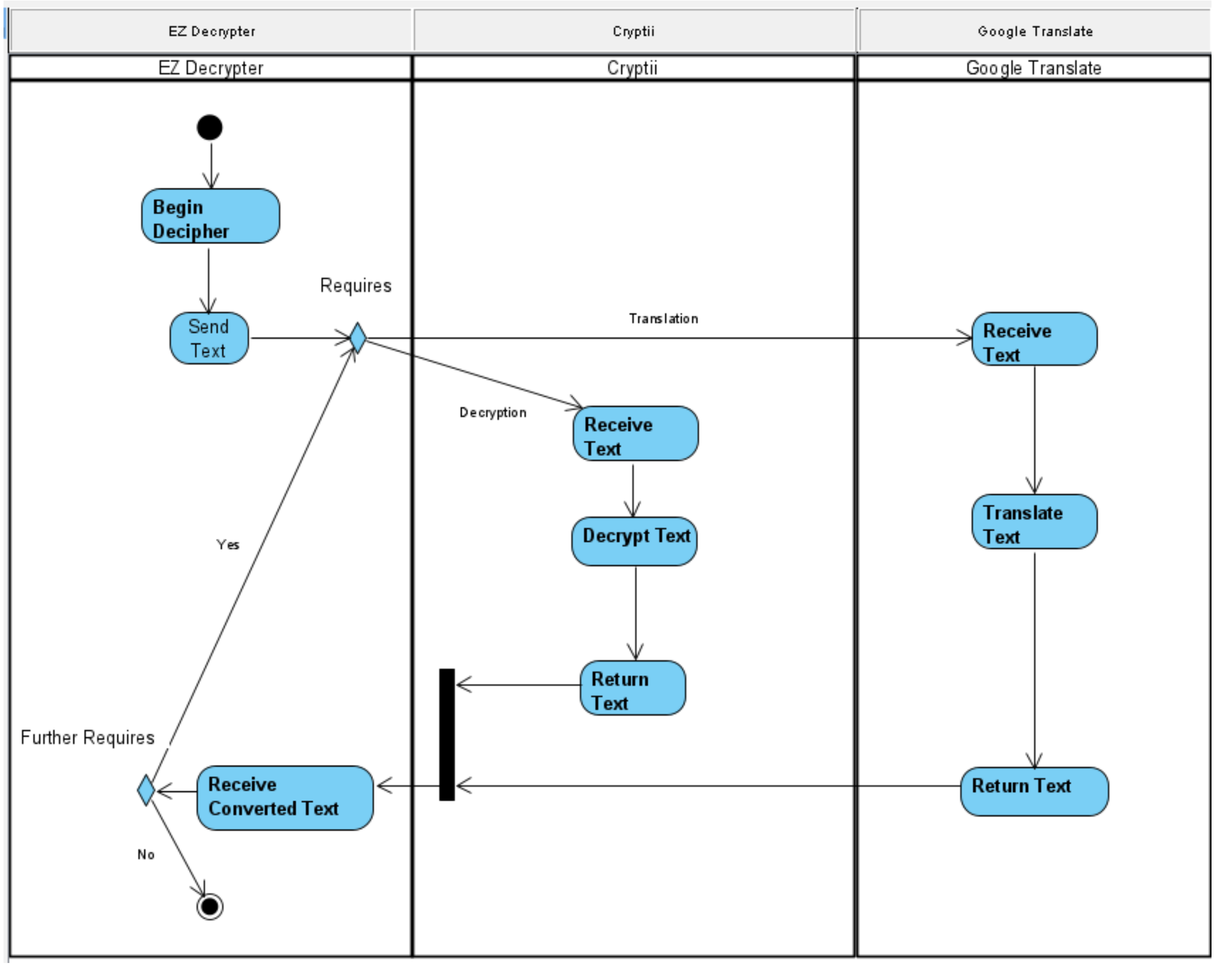
Figure 2: Image Conversion



Deciphering

Once the image has been converted to plain text and received by EZ Decrypter, it is then sent to either Cryptii, Google Translate, or both. Google Translate will receive the text and translate it based one the needs of the user. It will then return the text back to EZ Decrypter. Cryptii will have the same process but instead of translating, it will be encrypting or decoding. After completion of either process, should the text require further translation or decryption, it will be sent back through the process until it is completely deciphered. The deciphering process is shown in Figure 3.

Figure 3: Deciphering Process

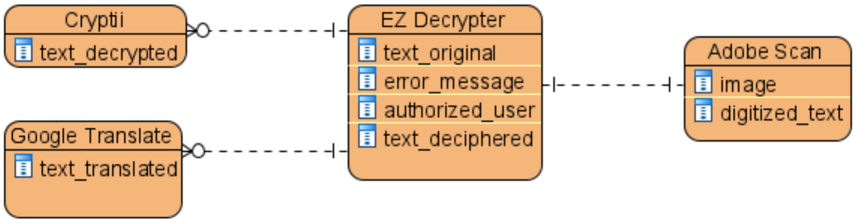


Data

Entity Relationship Diagram

The ERD for the data to be used and stored by EZ Decrypter, Adobe Scan, Cryptii, and Google Translate is shown in Figure 4.

Figure 4: ERD Diagram



Data Dictionary

The accompanying data dictionary for the ERD diagram is shown in Figure 5.

Figure 5: Data Dictionary

Field Name	Data Type	Data Format	Field Size	Description
text_original	String	Xxxxxxx	256	Original text gathered from Adobe Scan
error_message	String	Xxxxxxx	32	Text transcribed from capture in Adobe Scan
authorized_user	Boolean	T/F	1	Determines if the user can use the application
text_deciphered	String	Xxxxxxx	256	Deciphered text gathered from Cryptii and Google Translate
image	PDF			Image collected in Adobe Scan
digitized_text	String	Xxxxxxx	256	Text converted from PDF in Adobe Scan
text_decrypted	String	Xxxxxxx	256	Text decrypted from Cryptii
text_translated	String	Xxxxxxx	256	Text translated from Google Translate

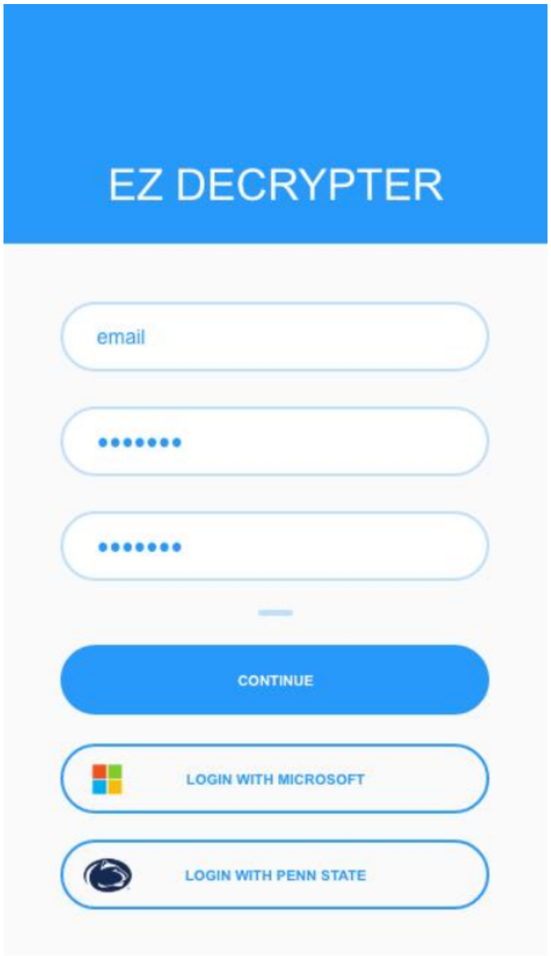
Prototype

The following prototype is the current application design that the team has created for EZ Decrypter.

Login Screen

Upon opening the EZ Decrypter application, the user will be greeted with a login page that will look similar to what is depicted in Figure 5. Once the user inputs their credentials, they are then validated through their agency before being approved for login.

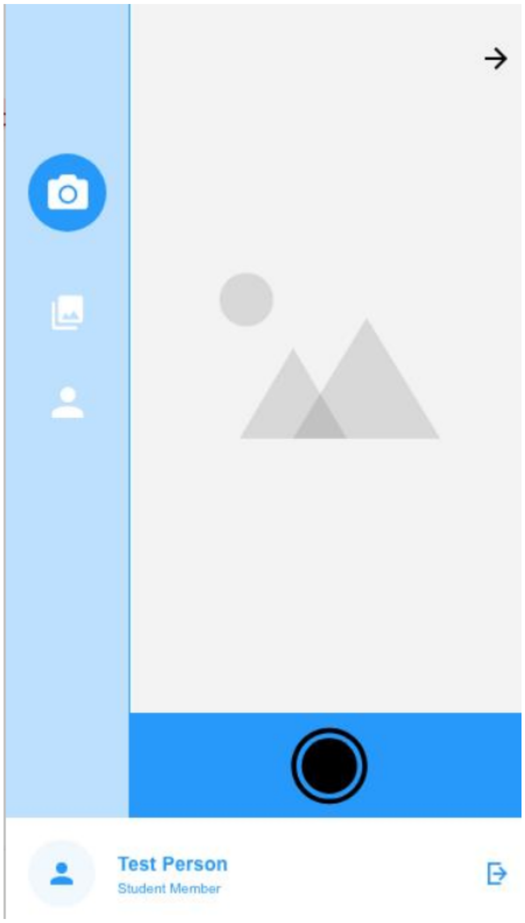
Figure 5: Login Screen



Main Screen

Once the user has successfully logged in, they will be greeted with a screen that will allow the user to either take a picture or upload one into the application. Adobe Scan will then take this image, extract the text, and digitize it in order for it to be encrypted/decrypted by Cryptii. This screen is shown in Figure 6.

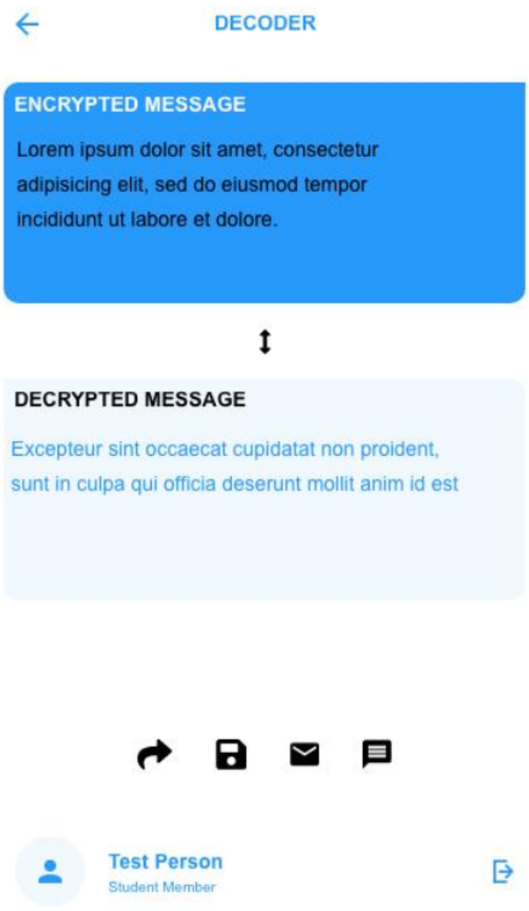
Figure 6: Main Screen



Encryption/Decryption Screen

Finally, once the image is taken/selected, the user will be redirected to the encryption/decryption page as shown in Figure 7. This page will display both versions of the text and allow the user to save it, email it, or return to the previous screen.

Figure 7: Encryption/Decryption Screen



ENCRYPTED MESSAGE

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore.

DECRYPTED MESSAGE

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est

Appendix C – Project Definition Report

Purpose

The goal of this project is to create a working prototype of a system that links together "off the shelf" technologies to decrypt handwritten encoded messages and deliver a simple-to-use, yet robust end-to-end solution.

Perhaps one of the biggest challenges faced by local law enforcement and national security agencies is effectively cracking the ciphertext used by the criminals. The process gets even more complicated when processing a handwritten ciphertext, as transcribing it to the system is time-consuming and error prone. Currently, available solutions in the market either lack critical functionality such as processing analog messages or are overly expensive and thus not feasible for every entity.

The project is designed to solve some of these critical problems, such as the hassle of manual transcription. This is accomplished by first digitalizing the handwritten encoded message and then decrypting it, effectively requiring minimal human interaction.

From the technology point of view, this project involves natural language processing, data security, and the linking of various information technologies (such as optical character recognition and readily available decryption software) over RESTful services. It is thus offering many options using a modular architecture for solving the problems faced by the deciphering agencies.

From the business point of view, the software architecture is designed to offer stable boundaries that are easier to work on independently. It allows end-users to extend the software and deploy the services modeled around their business domain. The power of extensibility enables agility in adopting new decryption algorithms, OCR technology, and other advanced functionality in the future with ease.

Economic feasibility is achieved by incorporating open-source software and other publicly accessible free third-party services, including Google Translation, Cloud Vision API, etc.

We plan to release the software free of cost, hoping that all mentioned disciplines would benefit substantially.

Stakeholders

The stakeholders have been identified from the software engineering point of view based on their functionality within the domain.

Internal stakeholders are as follows,

1. **Project Sponsor -** Lawrence Dupak. Primarily concerned with ensuring that the project delivers the agreed-upon business benefits.
2. **Project Leader -** Makes the final design decision as to the development of the system and acts as a chief product architect.
3. **Team Members -** Contribute code to the codebase and act as product architects.

External stakeholders are as follows,

1. **Plug-In Providers -** Build custom modules and micro-services to improve or enhance the functionality programmatically.
2. **End-Users -** Utilize our system. Users include law enforcement agencies such as the FBI, NSA, etc. Additional users include other organizations that require a system of this nature, such as historians and archaeologists.
3. **Third-Party Solution Providers -** The providers of the "off the shelf" or open-source technologies whose systems are integrated in our product.

Scope/SOW

Scope Statement

Project Justification:

Law enforcement as well as many other organizations have difficulty with encrypting and decrypting handwritten messages. The project is intended to provide a solution to this problem by linking various technologies into one easily accessible and affordable system.

Project Characteristics and Requirements:

1. Link existing "off the shelf" or open-source technologies.
2. System will interpret or encode a handwritten message that may be written in a foreign language or "made up" alphabet.
3. System will digitize the handwritten message.
4. System will utilize linked technologies to decipher the message.
5. Creation/Submission of all deliverables for the project.
6. Creation of training manuals for users.

Summary of Project Deliverables

Project Management-Related Deliverables:

Group Contact and Leader Forms, Project Definition Report, Progress Reports, Evaluations. Fully functional prototype demonstrating the proposed system.

Product-Related Deliverables:

1. Solutions Alternatives Document
2. Rough Draft of Design Requirements Document
3. Design Requirements Document
4. Draft of Implementation Document
5. Implementation Document
6. User Training manuals.

Not in the Scope of the Project

1. Modification/reconstruction of technologies to be linked together.
2. Construction of software to decrypt the messages.

Project Success Criteria:

1. All deliverables are submitted on time.
2. Stakeholders are happy with the product.
3. System is functional, complete and solves the problem.
4. System is easily accessible, and costs close to $0.
5. Cost remains under $5000.

Operational Feasibility

The decryption system will provide organizations with the capability to take encoded messages and decrypt them to reveal their hidden meanings. This will provide organizations, such as those within law enforcement, the capability to defend/mitigate/identify/analyze/recognize threats and vulnerabilities as intel is gathered, or simply store data for future use.

This decryption system will be tailored to the user. A user will be able to launch the interface, upload a picture or a document, the system will then read that image or document using OCR technology, pulling the data, and copying it into plain text. The system then analyzes the text against various dictionaries, databases, and algorithms to decode the message. It will work in much of the same manner as Google's OCR technology.

Once the message has been decrypted, the data will be stored within the program. There, the organization can cross reference data and use the information as needed.

Benefits

This project will benefit all our stakeholders. Organizations like the FBI and NSA have been working on deciphering similar code for decades. Creating a system that will automatically decode handwritten encoded messages, typically in a foreign language or "made up" alphabet, will be incredibly helpful. The system we will create is to be very budget-friendly and relatively inexpensive to produce. The goal is not to exceed $5,000 for this project. We will rely heavily on using Commercial off-the-shelf (COTS) solutions and open-source products in order to be within the budget.

Open source has gained a lot of popularity and is used widely by large organizations. Using an open-source license will help this project promote collaboration, increase security, transparency, and lower total cost to operate. The cost of using our product will be as close to $0 as possible. The product will be accessible and flexible to all the customers and organizations.

Lastly, we will design a system that links technologies so that handwritten encoded messages can be digitized and processed by automated decryption techniques. This system will free up a lot of time for our primary stakeholders and anyone interested in using our product. The incorporated advanced technology will do most of the work itself to decrypt a message automatically.

Appendix D – Group Contract

Group Name: Group 3

Course/ Section: IST 440W, Section 002

Project Group Members Names and Sign-off

Name (Print)	**Primary Means of Contact Information (email, cell, etc.)**
Group Leader: Michael Gabrys	Mzg5796@psu.edu
Sara Cunningham	Slc5902@psu.edu
Yash Dube	ysd5014@psu.edu
Vera Gordienko	Tdg5106@psu.edu
Celina Vedro	cxv5076@psu.edu

Michael Gabrys is the group leader and will be responsible for:

- Managing the group project, beginning with a communication plan, topic selection and responsibility matrix. Creating a communication plan may be as simple as coordinating email or a group. He will also schedule meetings (live or virtual) and mediate member performance.
- Communicating with the instructor, including posting milestones.
- Managing any group issues that may arise such as group disfunction, missed deadlines, lack of participation, etc.

Removal of a Member

Members can be removed from the team for a variety of reasons. Prior to removal, the issues will be brought to the attention of the group and attempted to be resolved by the group leader. The group leader will communicate any and all actions taken with the group prior to conducting them. If a member is removed from the group, the instructor will be notified.

A group member may be removed for the following reasons:

- Failure to attend meetings.
- Failure to adequately participate in meetings, communication, or assignments.
- Plagiarism.
- Missing important deadlines that can severely affect the project.
- Submitting work that is deemed subpar or of poor quality.

Rules for Quality Work and Code of Conduct

The team will work together to produce quality content to the best of their abilities. Assignments will be reviewed by the team and submitted once the team agrees the work fulfills the assignment requirements and satisfies quality standards.

Team members will always conduct themselves in a professional manner, and in accordance with Penn State's Code of Conduct & Student Conduct Procedures Policy. All communications will be handled respectfully. The attempt to resolve any disagreement or dispute between individual team members shall first be made by those individuals. If a resolution is not achieved by the disagreeing parties, the issue will be addressed during the team's next weekly synchronous meeting.

All team members will adhere to the Penn State College of Information Sciences and Technology Academic Integrity Policy.

Participation

Adequate and prompt participation will be required of all group members.

Group members will:

- Attend all scheduled group meetings.
    - If a member cannot be present, they must inform the team at least 3 hours before the meeting or as soon as possible. The absent member is responsible for what he/she missed and is expected to follow up with group members (if needed). Numerous absences will be unacceptable and will be reflected in the group participation grade and/or cause for removal from the group.
- Be active and participate in group meetings and communications. Members will contribute to all discussions and assignments and put in their "fair share" of effort.
    - Lack of participation will be addressed by the group leader who will then handle the situation as deemed fit, communicating the actions with the rest of the group.

Division of Work

The work will be divided equally amongst group members so that all members actively contribute to the project.

We will:

- Assign work equally among all group members.
- Collaborate and communicate effectively about work being completed.
- Construct a schedule for assignments and draft deadlines to ensure everyone is on time with their work, checking in on progress weekly.
- Log work for every assignment or meeting. Each member is responsible for keeping a log of their work (attendance, meetings, part of assignments, etc.) during the semester.
- Review work as a group and address any issues as a team with the group leader moderating.

Communication

Communication is essential and required by all group members.

Members will:

- Review Teams communication daily.
- Respond within 24 hours.
- Attend all weekly group meetings.
- Inform the group of any issues as soon as possible.
- Share all documents in the files on Microsoft Teams.

Meeting Guidelines

All members are expected to attend all meetings unless prior notice of circumstances have been conveyed to the team. Group meetings will be conducted weekly to ensure that the team is staying on task and meeting deadlines. Members are expected to remain courteous and respectful during all meetings. Meetings will be productive, and all members are expected to actively contribute to all discussions.

Name	**Signature**	**Date**
Michael Gabrys	Michael Gabrys	9/30/2020
Sara Cunningham	Sara Cunningham	9/30/2020
Yash Dube	Yash Dube	9/30/2020
Vera Gordienko	Vera Gordienko	9/30/2020
Celina Vedro	Celina Vedro	9/30/2020

Appendix E – Progress Report Start of Analysis Phase

Summary

Since this is the first progress report, we as a group established the framework. The goal of the initial progress report is to primarily lay down a solid foundation for the project in the form of a collaborative document with input from the entire group. The project document is divided into five sections, which cover Tasks Completed, Tasks Started, Future Phases, Challenges, and Work Breakdown Structure. All the sections combined tell us how and in which direction the project will proceed as well as its status thus far.

We have completed brainstorming ideas and are currently researching technologies and the best agile methodology for this project. As a next step, we are planning to discuss our findings with the professor. We are facing a few challenges with scheduling meetings as members are across different time zones and schedules. We have adopted async meetings, and so far, we are very much satisfied with our progress. We are looking ahead to deliver a successful project.

Task Completed

- Group Contract
- Project definition report
- Draft of work breakdown structure

- Gantt chart created
- Two weekly group meetings – brainstorm ideas
- Studied popular code-breaking applications – tested a few existing applications to understand their behavior, functionalities, user-interface, and the technologies used

Task Started

- Research on COTS – compare and contrast COTS solutions
- Research on NonCOTS – develop a list of NonCOTS solutions
- Alternate solution document

Next Steps

- Mandatory group meeting with the professor
- Research on integration solutions
- Create proof-of-concept – a small project to test the technological feasibility

Challenges

One of the main challenges that we have confronted is that our group members have vastly different schedules. This makes establishing meeting times and communication somewhat difficult because usually we cannot get everyone in a meeting simultaneously. Another challenge is that not everyone has a significant amount of experience in various areas of the project. This poses an issue because then we must spend more time researching the problem or task at hand before we can proceed with solving it. These two challenges have been the main issues that we have come confronted so far, but we have been taking steps to try and solve these issues. For example, we now take meeting notes so if anyone is absent, they can catch up when they can. These notes can also be used for us to reference if need be. Also, we have begun delegating work for future deliverables sooner, this way we are not as likely to be surprised by the amount of work entailed.

Work Breakdown Structure

1. Initiation
    1. Group Contract
        1. Identify scope and project goals
        1. Identify technologies
    2. Project Definition Report
2. Planning
    1. Project Report Start of Analysis Phase
        1. Create schedule and project goals
        1. Analyze resources and budgets
        2. Establish processes
3. Execution
    1. Identify steps needed to achieve project goals

            1. Updates
                1. Project progress
                2. Evaluate performance
                3. Identify project progress

4. Monitoring
    1. Maintain status updates/SCRUM meetings
        1. Control scope creep
        2. Change processes on management as needed
        3. Q&A
        4. Track performance and goal achievements
5. Closing
    1. Identify lessons learned
        1. Project report
        2. Evaluations

Appendix F – Progress Report Analysis Phase

Summary

During our analysis phase, we have established and finalized our overall goals and understanding for how our team will solve the issue. We have determined the technologies that we will be using to achieve our project goals. We have begun developing low fidelity prototypes to further define the functions and interface of our application. Once we have concluded this phase, we will begin constructing high fidelity prototypes.

Since the first progress report, we have lost our project leader. As a result, Michael Gabrys has been assigned the role of project leader. New communication guidelines have been established along with weekly meetings. The team has written a new contract which has been signed by all members. This new contract lays new guidelines and expectations for team members as well as how various situations will be handled. Overall, our team has been able to establish greater cohesiveness which has led to much better communication and collaboration.

Task Completed

- Revised group contract
- Solutions alternatives document
- Revised work breakdown structure
- Updated Gantt chart
- Two group meetings/established new communication standards
- Researched online softwares/apps
- Researched COTS and NonCOTS
- Analysis phase

Task Started

- Gather COTS and NonCOTS
- Create an initial system using multiple COTS and NonCOTS
- Draft low fidelity prototypes
- Design requirements document

Next Steps

- Decide what COTS and NonCOTS to use
- Review prototypes
- Design UI
- Complete design requirements document
- Begin implementation document

Challenges

As stated in the previous progress report, our team members have a variety of schedules which leads to issues with real-time meetings and communication. What we have done is established a meeting time where almost all members can attend. During the meeting, notes are taken and then stored in the files of Microsoft Teams which is available to all team members. Additional meetings can be held with the project manager should any issues arise, or further clarification be required.

Another challenge has been the loss of our project manager. This issue has been resolved by Michael Gabrys being assigned to the position with adjustments being made to the overall communication procedures and group contract.

Additionally, we began our initial research on COTS and NonCOTS and gathered a variety of different options. However, there are many choices to choose from, so we will explore every possibility to understand what is necessary. Examining all of the apps/softwares will help narrow down our list to ensure that we create a practical solution within budget, satisfies users' needs, and is well integrated.

Overall, there are minor challenges we face such as defining the modules and processes that the application will use. However, we expect to resolve these issues upon further prototyping and development of our application.

Work Breakdown Structure

1. Initiation
    1. Group Contract

      1. Identify scope and project goals
         1. Identify technologies
   2. Project Definition Report
2. Planning
   1. Project Report Start of Analysis Phase
      1. Create schedule and project goals
         1. Analyze resources and budgets
         2. Establish processes
3. Execution
   1. Identify steps needed to achieve project goals
      1. Updates
         1. Project progress
         2. Evaluate performance
         3. Identify project progress
4. Monitoring
   1. Maintain status updates/SCRUM meetings
      1. Control scope creep
      2. Change processes on management as needed
      3. Q&A
      4. Track performance and goal achievements
5. Closing
   1. Identify lessons learned
      1. Project report
      2. Evaluations

Appendix G – Progress Report Implementation Phase

Summary

During our analysis phase, we have established and finalized our overall goals and understanding for how our team will solve the issue. We have determined the technologies that we will be using to achieve our project goals. We have begun developing low fidelity prototypes to further define the functions and interface of our application. Once we have concluded this phase, we will begin constructing high fidelity prototypes.

Since the first progress report, we have lost our project leader. As a result, Michael Gabrys has been assigned the role of project leader. New communication guidelines have been established along with weekly meetings. The team has written a new contract which has been signed by all members. This new contract lays new guidelines and expectations for team members as well as how various situations will be handled. Overall, our team has been able to establish greater cohesiveness which has led to much better communication and collaboration.

Task Completed

- Revised group contract
- Solutions alternatives document
- Revised work breakdown structure
- Updated Gantt chart
- Two group meetings/established new communication standards
- Researched online softwares/apps
- Researched COTS and NonCOTS
- Analysis phase

Task Started

- Gather COTS and NonCOTS
- Create an initial system using multiple COTS and NonCOTS
- Draft low fidelity prototypes
- Design requirements document

Next Steps

- Decide what COTS and NonCOTS to use
- Review prototypes
- Design UI
- Complete design requirements document
- Begin implementation document

Challenges

As stated in the previous progress report, our team members have a variety of schedules which leads to issues with real-time meetings and communication. What we have done is established a meeting time where almost all members can attend. During the meeting, notes are taken and then stored in the files of Microsoft Teams which is available to all team members. Additional meetings can be held with the project manager should any issues arise, or further clarification be required.

Another challenge has been the loss of our project manager. This issue has been resolved by Michael Gabrys being assigned to the position with adjustments being made to the overall communication procedures and group contract.

Additionally, we began our initial research on COTS and NonCOTS and gathered a variety of different options. However, there are many choices to choose from, so we will explore every possibility to understand what is necessary. Examining all of the apps/softwares will help narrow down our list to ensure that we create a practical solution within budget, satisfies users' needs, and is well integrated.

Overall, there are minor challenges we face such as defining the modules and processes that the application will use. However, we expect to resolve these issues upon further prototyping and development of our application.

Work Breakdown Structure

1. Initiation
   1. Group Contract
      1. Identify scope and project goals
         1. Identify technologies
   2. Project Definition Report
2. Planning
   1. Project Report Start of Analysis Phase
      1. Create schedule and project goals
         1. Analyze resources and budgets
         2. Establish processes
3. Execution
   1. Identify steps needed to achieve project goals
      1. Updates
         1. Project progress
         2. Evaluate performance
         3. Identify project progress
4. Monitoring
   1. Maintain status updates/SCRUM meetings
      1. Control scope creep
      2. Change processes on management as needed
      3. Q&A
      4. Track performance and goal achievements
5. Closing
   1. Identify lessons learned
      1. Project report
      2. Evaluations