Appendix I-MCC Installation and User Guide

1 Objectives

The objective of this document is to provide an overview of Intelligent Malicious Content Checker (I-MCC) application and the necessary information to use the application. The manual assumes that the reader has sufficient understanding on system implementation, Robotic Process Automation (RPA) , programming language (Python), Google Cloud API and FlaskApp.

2 Scopes

The high-level scope of the user guide will encompass THREE (3) sections:

1. System Overview
2. Installation and Configuration
3. User Manual (Use Case)

3 System Overview

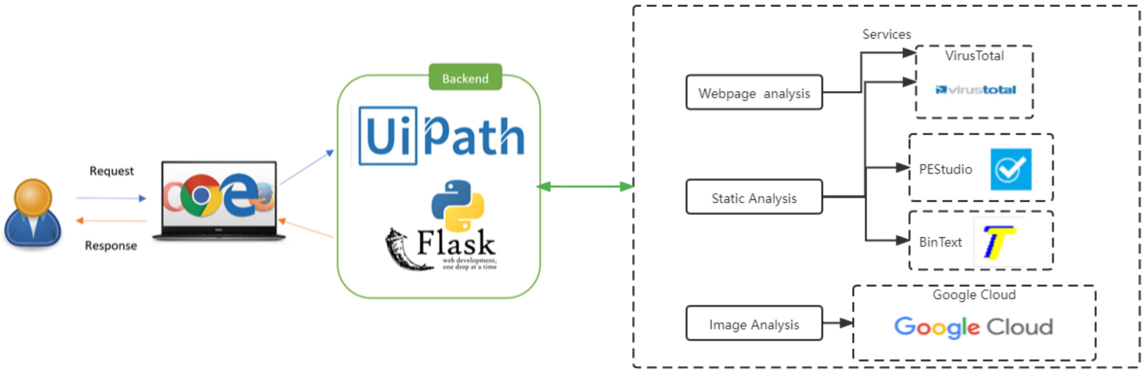


Figure 1 System Overview

Figure 1 shows the system overview for I-MCC. Following are the components used in I-MCC.

1. Web browser – User interface with I-MCC websites
2. UIPath – Robotic Process Automation tools to automate the process
3. FlaskApp – tools, libraries and technologies that are used in this project to build a web application
4. Google Cloud API – Vision API for detecting inappropriate image contents
5. VirusTotal, PEStudio and BinText – Tools and websites to perform static malware analysis

4 Installation

4.1 Requirements

Description	Specification
Software	• Python3.8 • UIPath Community Edition
Packages	All required python packages are defined in the requirements.txt, please use 'pip install' command to install them.

4.2 Backend Server Setup and Configuration

Below shows the step to setup the backend server. It only requires minimum setup and configuration as all the necessary codes are included in the github folder – SystemCodes.

1. Install UIPath Community Edition

   https://docs.uipath.com/installation-and-upgrade/docs/studio-install-studio

2. Use 'git clone' command to download the project from the following URL:

   https://github.com/mediana-medy/ISA-IPA-2021-11-17-IS03FT-GRP1-IntelligentMaliciousContentChecker_I-MCC

3. All the required codes for UIPath, FlaskApp, Google Cloud API, and tools used in the I-MCC are resided in SystemCodes folder

   a) UIPath/StaticAnalysis – contains all the required RPA XAML files, static analysis tools like BinText, and PEStudio. Please ensure the downloaded files containing the following directory and RPA files.

Names	Directory/Files
back_up_malware-zip_file	Directory
done_analyse	Directory
malware_beware	Directory
pythonscript	Directory
tools	Directory
virustotal_result	Directory
resultfile	Directory
MainStaticA.xaml	RPA Files
BinText.xaml	RPA Files
CleanUpFile.xaml	RPA Files
DecryptFile.xaml	RPA Files
PEStudio.xaml	RPA Files
project.json	RPA Files
QueryVirustotalHash.xaml	RPA Files

QueryVirustotalURL.xaml	RPA Files
StaticAnalysis.xaml	RPA Files

b) Download_folder – contains the results of analysis for users to download

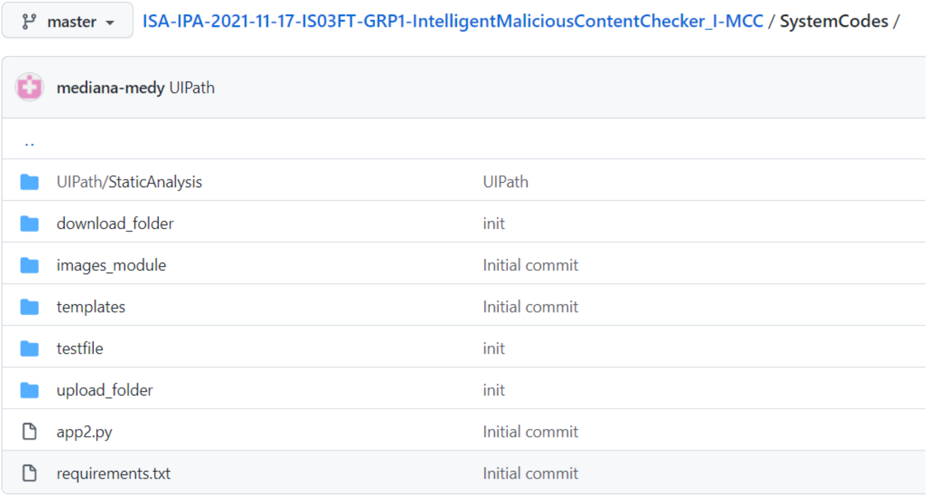
c) Images_module – contains the python files for performing the images content checker using Google Cloud API.

d) Templates – contains two html pages, ie upload.html and download.html

e) Testfile – contains two zipped file that users can use for testing the images content checker module and static analysis checker.

f) Upload_folder – contains the zipped file that are uploaded by users for performing analysis

g) App2.py – main python files for receiving the request from users and calling relevant functions to perform the three modules, ie URL Checker, Images Content Checker and Files Checker (Static Analysis)



ISA-IPA-2021-11-17-IS03FT-GRP1-IntelligentMaliciousContentChecker_I-MCC / SystemCodes /

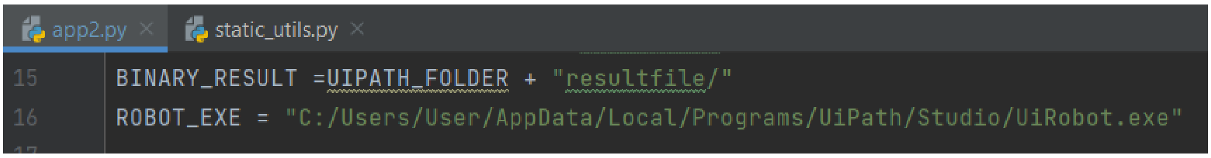
mediana-medy UIPath

..

UIPath/StaticAnalysis	UIPath
download_folder	init
images_module	Initial commit
templates	Initial commit
testfile	init
upload_folder	init
app2.py	Initial commit
requirements.txt	Initial commit

Figure 2 System Codes (Folder)

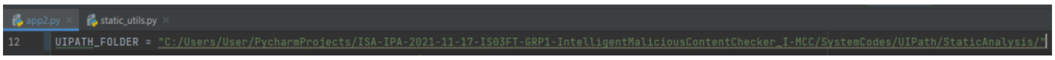
4. After installation of UIPath, ensure that UIRobot.exe are also installed and resided in this default folder: *"C:/Users/User/AppData/Local/Programs/UiPath/Studio/UiRobot.exe".* Please update the ROBOT_EXE in app2.py if the UIRobot.exe is not in the default folder.



```
BINARY_RESULT =UIPATH_FOLDER + "resultfile/"
ROBOT_EXE = "C:/Users/User/AppData/Local/Programs/UiPath/Studio/UiRobot.exe"
```

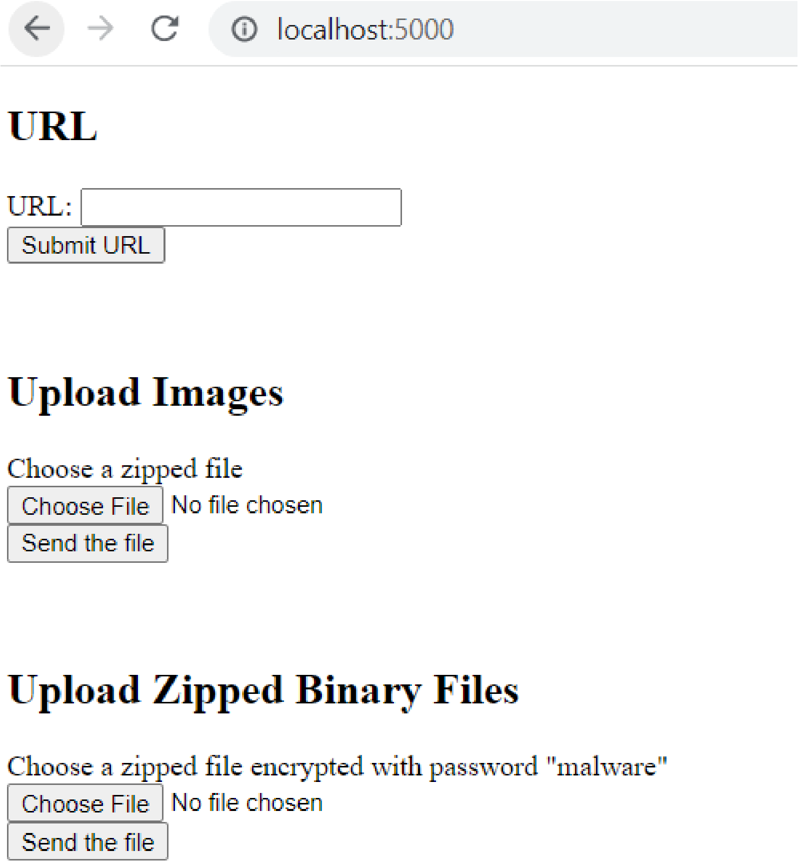
Figure 3 Define ROBOT_EXE in app2.py

5. Please update the UIPATH_FOLDER in app2.py where the GitHub downloaded files reside according to your environment.



```
UIPATH_FOLDER = "C:/Users/User/PycharmProjects/ISA-IPA-2021-11-17-IS03FT-GRP1-IntelligentMaliciousContentChecker_I-MCC/SystemCodes/UIPath/StaticAnalysis/"
```

6. Run the app2.py and click browser http://localhost:5000



URL

URL:

Upload Images

Choose a zipped file

Upload Zipped Binary Files

Choose a zipped file encrypted with password "malware"

Figure 4 I-MCC Webpage

7.

5 User Guide

5.1 Use Case Overview

I-MCC provides 3 modes for users, URL Checker, Images Content Checker and File Checker (Static Malware Analysis).

5.1.1 URL Checker

Input URL in the textbox and click Submit URL for checking the URL in VirusTotal dataset whether it is marked as malicious in multiple antivirus engine.

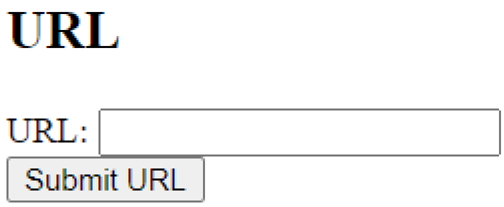
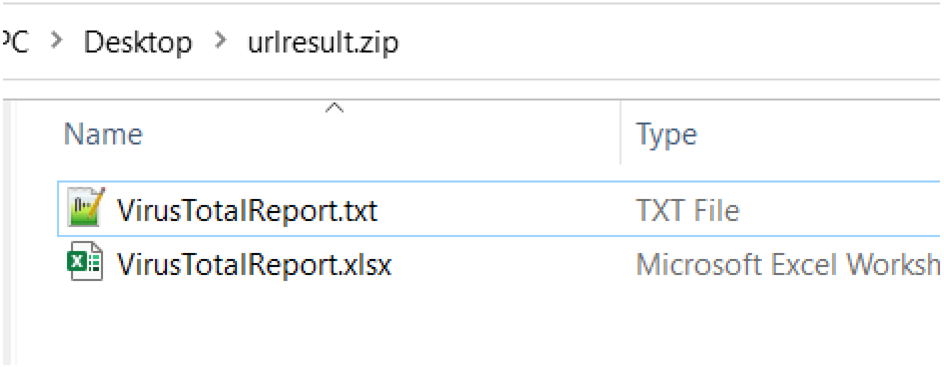


Figure 5 URL Input

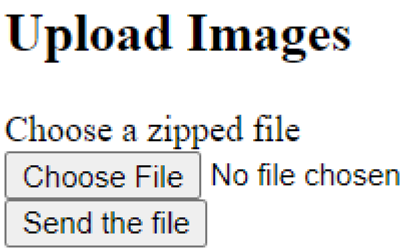

C › Desktop › urlresult.zip

Name	Type
VirusTotalReport.txt	TXT File
VirusTotalReport.xlsx	Microsoft Excel Worksh

Figure 6 URL Results - VirusTotal Report

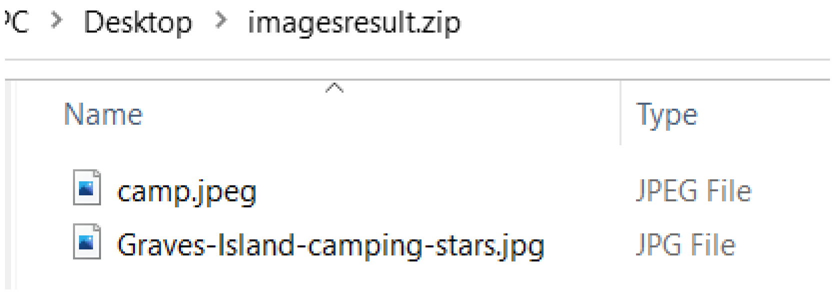
5.1.2 Images Content Checker

Upload images in zip and click “Send the file” for checking whether the zip file contains any inappropriate images like violence, and abuse. It will delete all the inappropriate images. A zip file with all the inappropriate images is deleted will be returned to users for download.


Upload Images

Choose a zipped file

Figure 7 Images Content Checker (Input Images)

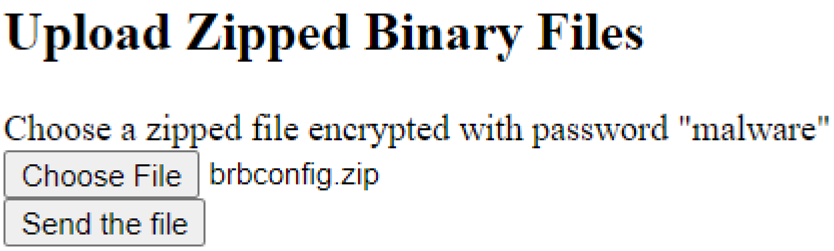


Name	Type
camp.jpeg	JPEG File
Graves-Island-camping-stars.jpg	JPG File

Figure 8 Images Content Checker Output contains only appropriate images

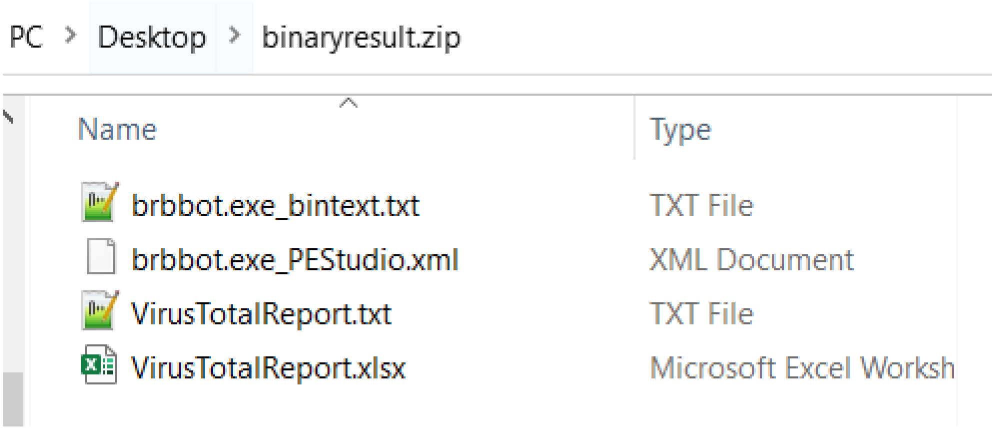
5.1.3 File Checker

Upload binary files in zip and encrypted with password “malware” for performing static malware analysis. It will perform analysis as stated in Section XX Static Analysis then output PEStudio results, BinText results and VirusTotal Hash Check results for users to download.


Upload Zipped Binary Files

Choose a zipped file encrypted with password "malware"

Figure 9 Files Checker - Static Malware Analysis (Input Password-Protected Zip Files)



PC > Desktop > binaryresult.zip

Name	Type
brbbot.exe_bintext.txt	TXT File
brbbot.exe_PEStudio.xml	XML Document
VirusTotalReport.txt	TXT File
VirusTotalReport.xlsx	Microsoft Excel Worksh

Figure 10 Files Checker - Static Malware Analysis containing BinText, PEStudio and VirusTotal results

For ISS Use Only		
Programme Name:	**Project No:**	**Learner Batch:**
Accepted/Rejected/KIV:		
Learners Assigned:		
Advisor Assigned: Contact: Mr. GU ZHAN / Lecturer & Consultant Telephone No.: 65-6516 8021 Email: zhan.gu@nus.edu.sg		