**SmartEye Technical MANUAL**

*SmartEye Data Extractor Tool*

Prepared by: David Gray

May, 2017

**USER'S MANUAL**

**TABLE OF CONTENTS**

Page #

[1.0 GENERAL INFORMATION](#__RefHeading___Toc4234_2036400096)

[1.1 System Overview 1](#__RefHeading___Toc4236_2036400096)

[2.0 Setting up SmartEye](#__RefHeading___Toc4238_2036400096)

[2.1 Required resources](#__RefHeading___Toc4240_2036400096) 2

[2.2 Running the program for the first time](#__RefHeading___Toc4242_2036400096) 2

[2.2 Running the project](#__RefHeading___Toc4572_2036400096) 2

[2.3 Uploading image files to SmartEye](#__RefHeading___Toc4244_2036400096) 2

[2.4 Uploading pdf files to SmartEye](#__RefHeading___Toc4574_2036400096) 2

[2.5 Retrieving database information](#__RefHeading___Toc4576_2036400096) 3

[2.6 Running unit tests](#__RefHeading___Toc4578_2036400096) 3

[2.7 Changing database connections](#__RefHeading___Toc4580_2036400096) 3

[2.8 Changing Google API Key](#__RefHeading___Toc4582_2036400096) 3

**1.0 GENERAL INFORMATION**

# GENERAL INFORMATION

## 1.1 System Overview

The SmartEye Data Extractor tool is a Golang based multithreaded project that is capable of; multiple file uploading and storing, sending and receiving data requests from a web application and Google Cloud Vision, and performing file splitting and conversion.

The SmartEye Data Extractor tool consists of: an API to be used as a middleman between the web application and the main Golang application; a Dex file object that will be created and used for each transaction request on the web application to the main application; a file splitter/converter; a database store and connection; and a Google Cloud Vision service/connection.

The web application for the SmartEye Data Extractor tool uses HTML5, CSS3, and jQuery for webpage creation, to display webpages, and to respond to user/client requests.

**2.0 SYSTEM SUMMARY**

# Setting up SmartEye

## 2.1 Required resources

To run the SmartEye Data Extractor tool you must have:

* Gogland Golang IDE
* SQLBoiler External Library
* Gin Gonic External Library
* DBVisualizer
* An internet connection

These resources must be installed/running for the tool to work.

## 2.2 Running the program for the first time

To run the SmartEye Data Extractor tool for the first time, you must create a new Gogland project and

copy all the SmartEye data contents into this project. In the settings option for Gogland, you must

navigate to the database connections settings and set up a database connection and host that the extractor

will connect and store data to. From there, right click on the file in the SQL folder named tables.sql and

run it. This will delete all current data in the database, and start the database server connection. Once thai

is done, click the build and run button in Gogland to run the project.

## 2.2 Running the project

To run the project, make sure your database connection has been established by executing the tables.sql

file in the SQL folder, and then click the build and run button in Gogland.

## 2.3 Uploading image files to SmartEye

To upload image files to SmartEye you can access the web application through either localhost:8080 for

testing or https//smarteye.redeye.co/access/db for release.

To upload image files to the SmartEye database, locate the “IMAGE FILE UPLOAD” section on the web

page. Click on the button below that section that says “ADD FILES”. A pop up box should appear

showing your computer’s file directory. Navigate and select any image files you wish to upload and then

hit okay. Once that is done, hit the button that says “START UPLOAD”. Your files will now start

uploading.

If an error should occur during the upload process, a pop up box will appear stating that an error has

occurred, telling you what the error was, and a potential solution to the problem.

If no errors occur, your file will have been uploaded to SmartEye’s database.

## 2.4 Uploading pdf files to SmartEye

To upload pdf files to SmartEye, locate the “PDF FILE UPLOAD” section on the web page. Click on the button below that section that says “ADD FILES”. A pop up box should appear

showing your computer’s file directory. Navigate and select any image files you wish to upload and then

hit okay. Once that is done, hit the button that says “START UPLOAD”. Your files will now start

uploading.

If an error should occur during the upload process, a pop up box will appear stating that an error has

occurred, telling you what the error was, and a potential solution to the problem.

If no errors occur, your file will have been uploaded to SmartEye’s database.

## 2.5 Retrieving database information

To make requests for data in the SmartEye database, locate “SEARCH BAR” section on the web page. In

the search bar below the title, enter a file name or company name to retrieve data for that file or company.

Click the “SEARCH” button to make the search request. Data relating to your search request will be

returned and displayed on the web page.

## 2.6 Running unit tests

To run unit test files, find files that end with \_test.go and find the function that you would like to test. Right click on that function and click run test or run test with coverage. If you run the test it will return a pass or fail, if you run the test with coverage it will return a pass or fail and if it passed, it will tell you how much code coverage the function tested and passed.

## 2.7 Changing database connections

To change the database connection that the program will connect to, locate the main.go file. Within the main.go file there is a function called initDB which sets up the database to connect to. Change any relevant information as required to connect to the new database you which to use.

## 2.8 Changing Google API Key

To use a different Google API Key, find the config.yml file in the Config folder and change the APIKey value to your new API key that you want to use.