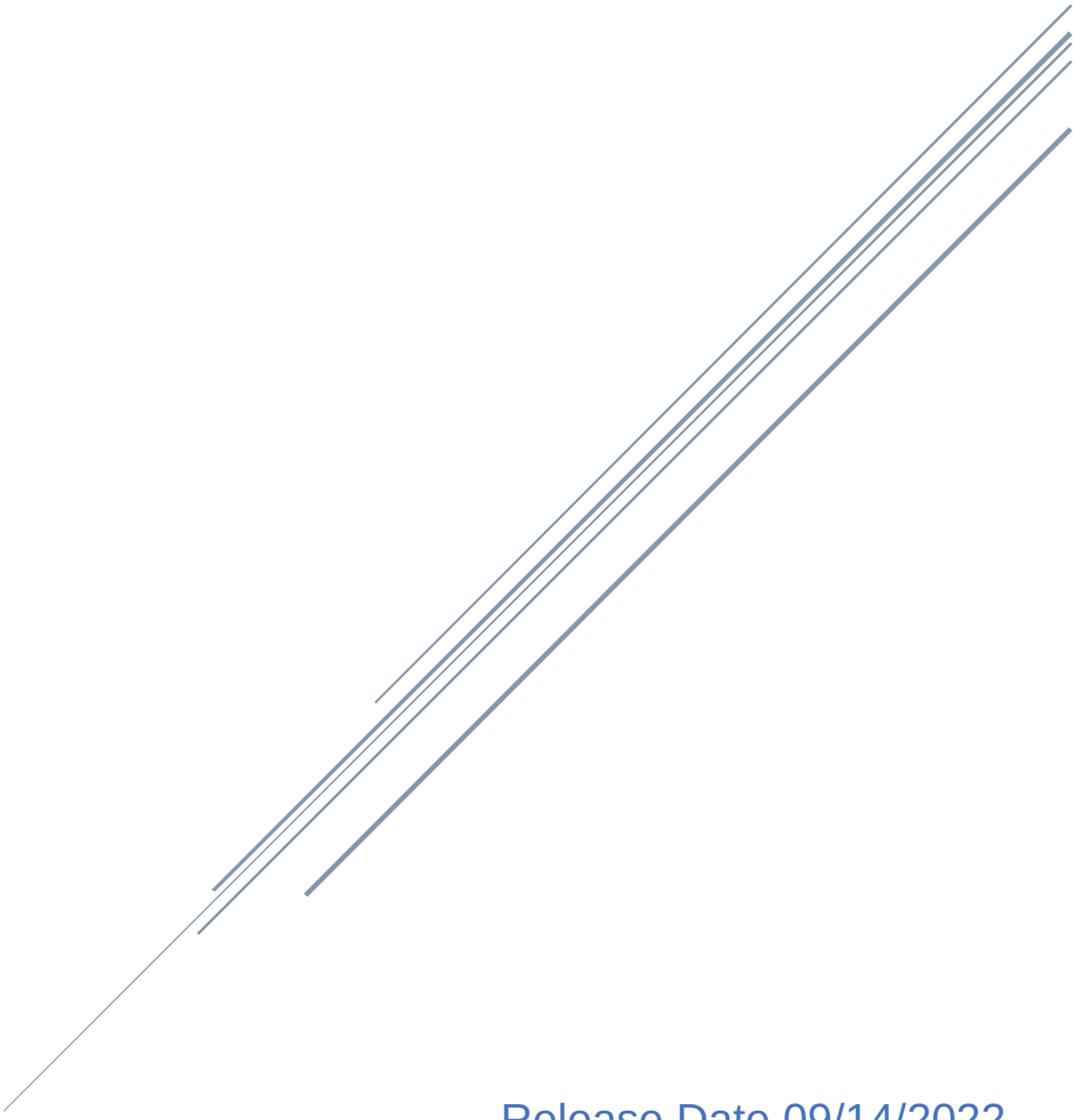
RAPID ADDRESS DE-IDENTIFIER (RAD) TOOL

National Environmental Public Health Tracking Network



Release Date 09/14/2022
Version 1.2

VERSION HISTORY

Version #	Author	Revision Date	Reason
1.0	Kent Nardin	7/21/2022	Initial Draft

TABLE OF CONTENTS

Section 1 – Overview

The Rapid Address De-identifier (RAD) Tool is a Java-based desktop application designed to remove sensitive address information from an input file and create a new file where the address information is replaced with a universally unique identifier (UUID). The RAD Tool also generates geocoding files that can be used for the geocoding process with Texas A&M University.

This document outlines the steps required to install and run the RAD Tool, how to use the tool, files that are generated, and common errors.

Section 2 – System Requirements

The RAD Tool was developed and tested to work on machines using a Windows operating system. For the tool to run, Java 1.8 or higher must be installed on the machine.

Section 3 – Installation & Running

1. Download “RAD.jar” from the GitHub repository.

2. Move the RAD.jar file to the desired location on the machine.

3. To run the RAD Tool, simply double-click the RAD.jar file. An alternative running method is to open a Command Prompt in Windows and execute the command, “java -jar RAD.jar.”

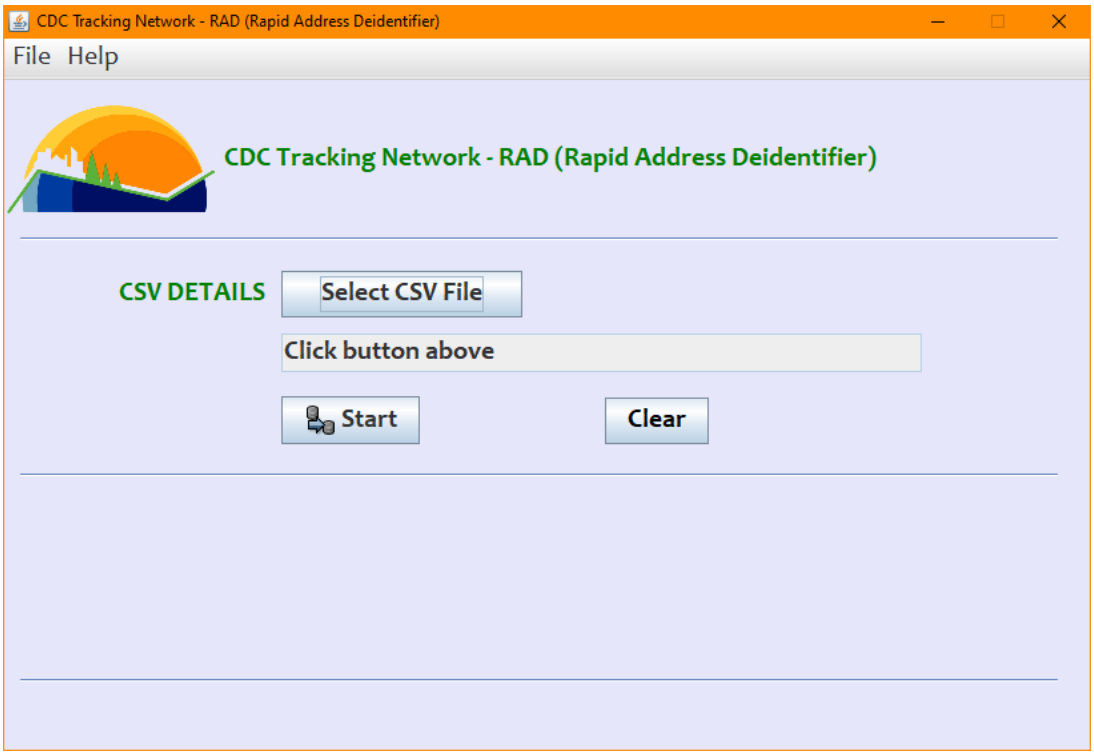


Figure 1: RAD Tool

Section 4 – Input File

The RAD Tool utilizes comma-separated values (CSV) files as input. The tool will display a warning popup if a different type of file is selected for processing. Remember these few notes about the CSV input files

- The first line in the CSV file must be the header row.
  - Required header fields: RowIdentifier, Street, City, State, ZIP, AddressIdentifier
  - Optional header fields: Company, Phone, GrantDate, Activities, DBAs

- Double quotes must surround any data that contains a comma.
  - Ex: Correct: “walking, hiking, running” Incorrect: walking, hiking, running

- Any data for the AddressIdentifier field must be the entire address string and enclosed by double quotes.
  - Ex: “1234 Main St., Sometown, OH, 43210”

Section 5 – How to Use

1. Begin by running the RAD Tool.

2. Click the “Select CSV File” button, navigate to the desired file, and click the “Open” button.
    a. **Note**: The output files generated by the RAD Tool will be created in the same directory as the input CSV file.

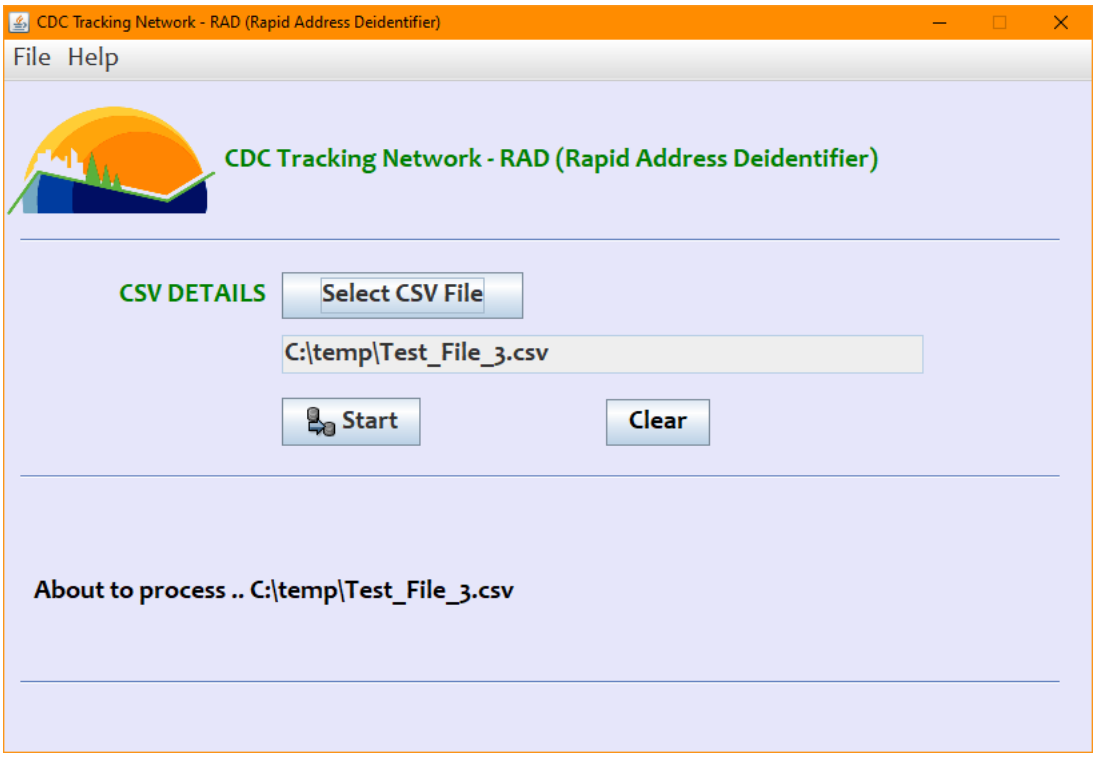


Figure 2: Ready to begin processing

3. Click the “Start” button to begin processing. You will know the tool is done processing when the message, “Successfully completed the processing!!!” displays in the main window. If the tool encounters an issue during processing, “An error occurred. Please check the RAD.log file.” will instead display in the main window. Log files will be covered in Section 7.

4. To close the RAD Tool, click on the “X” button in the top right corner or select “File” → “Exit” from the top left corner.

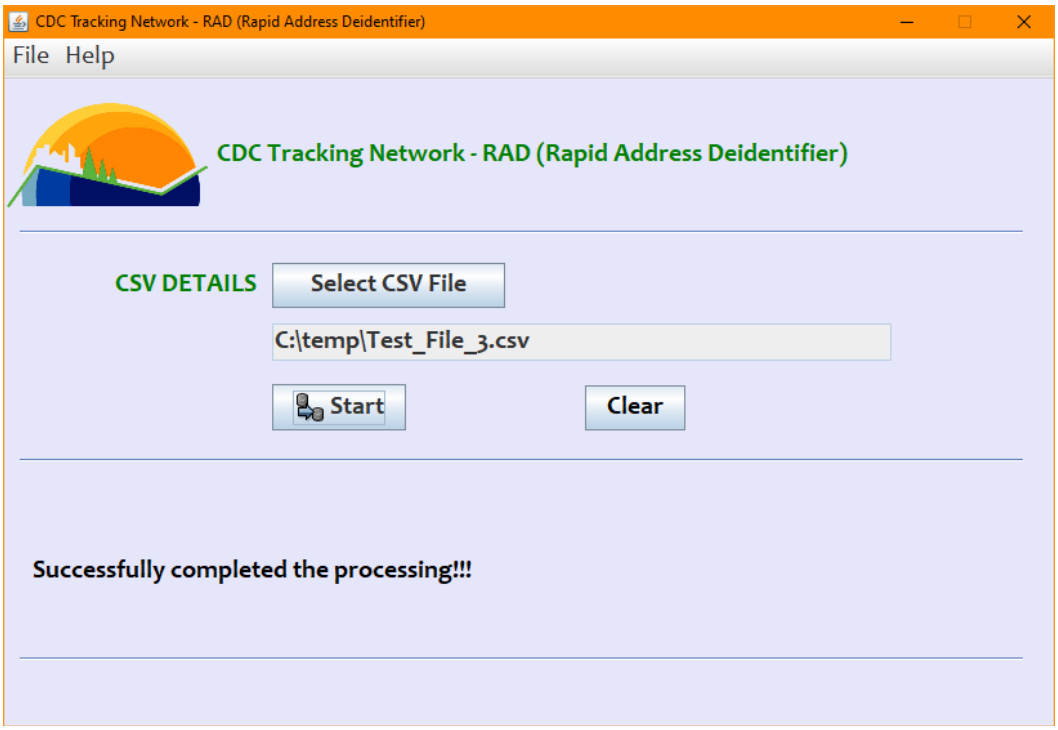


Figure 3: Successful processing

Section 6 – Output Files

During execution, the RAD Tool creates a new folder named “PROCESSED” and places all the output files into this folder. This PROCESSED folder is in the same directory as the input CSV file. Output files will all begin with the name of the input CSV file to help distinguish output files from different input files.

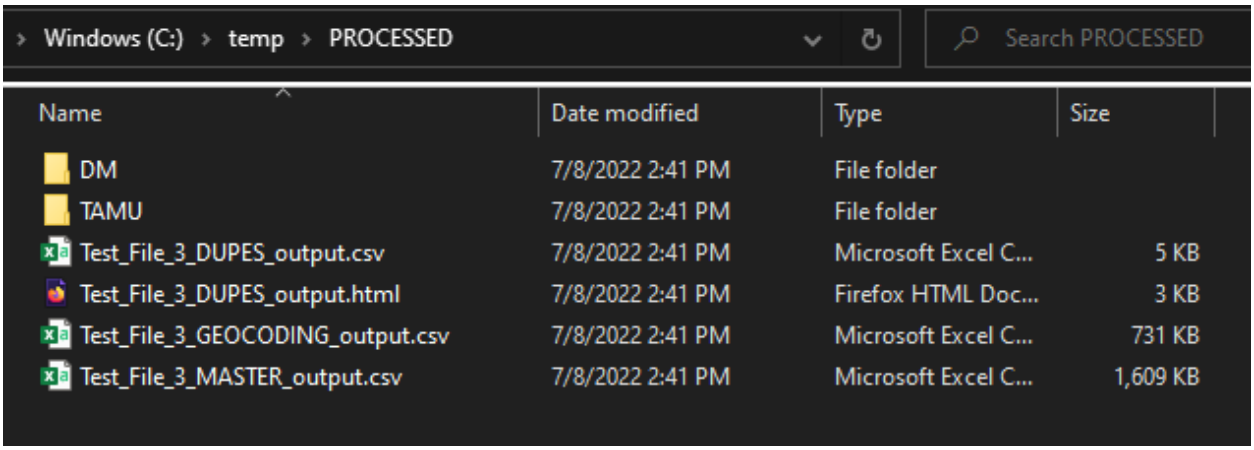


Figure 4: PROCESSED output folder

The RAD Tool generates four types of output files:

1. Data Management (DM) output file (addresses replaced with universally unique identifiers (UUIDs))
    - This file is located inside the “DM” folder, which is inside the PROCESSED folder, and has the file name, “[input file name]_DM_output.csv.” Ex: “Test_File_3_DM_output.csv”
    - This is the main output file. All address fields have been removed, and each address has been assigned a UUID.
    - This file can be shared with the DM team and Battelle.

2. Duplicates (DUPES) files - CSV and HTML versions
    - These files are named “[input file name]_DUPES_output.csv” and “[input file name]_DUPES_output.html.”
    - The CSV version provides a count of duplicate addresses in the input CSV file. The addresses are represented as their UUIDs in these output files.
    - The HTML version provides a list of output files as well as metrics for addresses, records, and processing time.

3. MASTER file
    - This file is named “[input file name]_MASTER_output.csv.”
    - The MASTER file is a copy of the input file with the UUID data appended to each row and is how the user can identify which UUID is associated with which address.

4. GEOCODING files
    - All the GEOCODING files contain the same information: UUID, AddressIdentifier, and specific address fields.
    - There are two types of GEOCODING files.
        - A single file named, “[input file name]_GEOCODING_output.csv.” In this file is a row for every row in the input file.
        - In the “TAMU” folder, the above GEOCODING output file is broken into 5MB sized files. This is to conform with the file size constraints by TAMU. There will always be at least one output file in this folder. The file names will be appended with numbers starting with zero. Ex: “Test_File_3_GEOCODING_output_**0**.csv,” “Test_File_3_GEOCODING_output_**1**.csv,” etc.

Section 7 – Log File

When the RAD Tool executes, it creates a “RAD.log” file within a “log” folder located in the same location as the RAD.jar file. This log file contains messages generated by the tool as it processes input files as well as messages regarding any errors encountered.

Note: When the RAD Tool is run and a file is processed for the first time in a day, the tool will automatically create a log file for the date the RAD Tool was the last run. For example, if the tool was run on March 1st and again on March 4th, upon processing a file on March 4th, a log file for March 1st would be created. This file would be named, "logsRAD-info-error-2022-03-01.log." The purpose of this logging is to create a historical record of previous executions.

Section 8 – Common Errors

Many errors encountered during the processing of an input CSV file will be a result of issues with the data in the input file. For these common errors, the RAD.log file will include a "**TO FIX**" statement that provides some guidance on how to fix the error that was encountered. Below are some common data issues that will cause errors during processing.

- Empty input file

- Header row
  - Missing header row
  - Missing or misspelled header fields
    - A missing header field may also have missing data, so both should be investigated if this error occurs.

- Data rows
  - Header row present but no data
  - Missing data for a required field
    - Ex: The header row fields are First Name, Last Name, and Occupation, (First and Last Name are required). The below data row would cause an error (note the missing data between the two commas).
    - John,,Chef
  - Mismatch between the number of header fields and the number of data fields in a row
    - This is different than the above error in that both data and comma(s) are missing from a row. Data containing commas but not enclosed in double quotes can cause this error.

Section 9 – Important Notes

- The RAD Tool does not check the validity of an address. For example, it does not check the spelling of a street name.

- Files with the same address but have slight differences in the address will yield different UUIDs in the output file. For example, if one address contained “St. Paul” as the city while another contained “Saint Paul,” this would result in different UUIDs.

- Rerunning the RAD Tool with the same input file will result in different UUIDs for the addresses in the output file. Therefore, it is important not to lose output files that are then sent for geocoding, as these output files cannot be recreated by the RAD Tool.

Section 10 – Questions

Please direct any questions regarding this document or the RAD Tool to nephtrackingsupport@cdc.gov.