

**Functional Specification Document**

**E-Challan**



**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Purpose | Author | Reviewer | Release Date |
| 1 | Initial Document | Ashwini Khandekar  Bhakti Deshmukh | Parimal Gujarathi | 24th Jan 2023 |

**Table of Contents**

[**1.** **Introduction/Background** 4](#_Toc94547184)

[**2.** **Functional Requirements** 5](#_Toc94547185)

[**2.1** **Business Need** 5](#_Toc94547186)

[**2.2** **High Level Requirements** 5](#_Toc94547187)

[**3.** **Functional Workflow** 6](#_Toc94547188)

[**3.1** **High-Level Diagram** 6](#_Toc94547189)

[**3.2** **Detailed Functional Requirements** 7](#_Toc94547190)

[**4.** **Key Assumptions & Dependencies** 13](#_Toc94547191)

[**5.** **Inputs** 14](#_Toc94547192)

[**5.1** **Prerequisites** 14](#_Toc94547193)

[**5.2** **Inputs Required** 14](#_Toc94547194)

[**6.** **Error Handling** 15](#_Toc94547195)

[**7.** **Out of Scope for RPA** 15](#_Toc94547196)

# **Introduction/Background**

The Functional Specification Document is a document that provides detailed information on how the system solution will function. The purpose of this document is to highlight the client requirements to find challan details. In this document will be the detailed requirements including system inputs and outputs, process flows and diagrams.

# **Functional Requirements**

## **Business Need**

* Currently, client is navigating to e-challan payment Maharashtra page and then client navigates to the vehicle number or challan number manually. The client manually enters the vehicle number, chassis/engine number and challan number and click the captcha. Then client click the submit button manually. Client fetch the e-challan details and enters the e-challan details in output file and sort the details on the basis of challan amount.
* This is a manual process and require human to spend considerable time in finding challan details every time. Client wants to automate this manual process by Robotic Process Automation. By automating this process, Automation makes this process faster, scalable and less error prone.

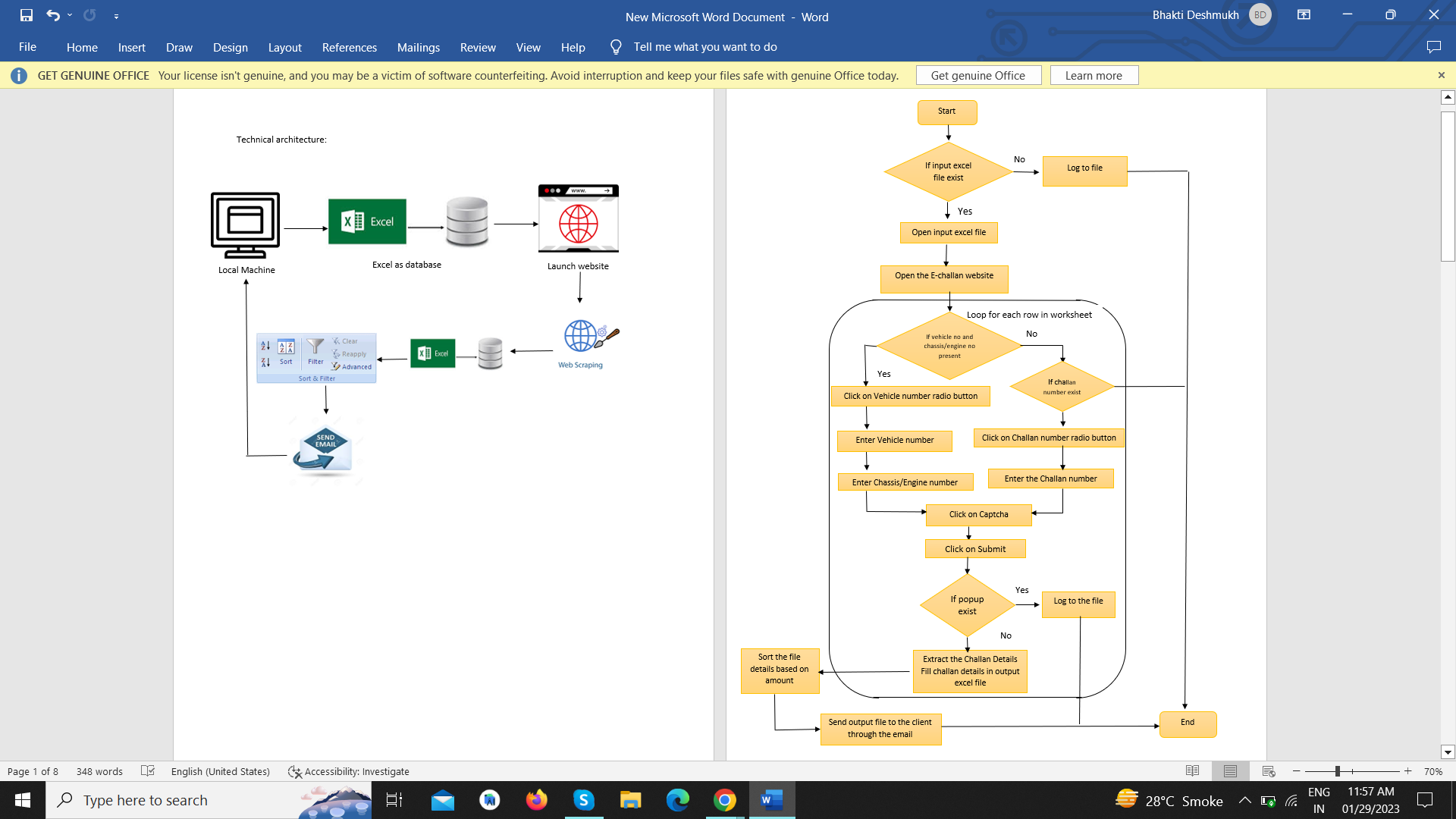
## **High Level Requirements**

The high-level steps to perform during the process are as follows:

1. Navigate to website- https://mahatrafficechallan.gov.in/
2. Select vehicle number or challan number.
3. Selected vehicle number then enter vehicle number and last four digit of chassis/engine number.
4. Selected challan number then enter challan number.
5. Click on re-captcha.
6. Click on submit button.
7. Read below-mentioned challan details and enter it in output file.
   1. Amount
   2. Towing Amt. + GST (18%)
   3. Challan No.
   4. Challan Date & Time
   5. Driver Name
   6. Vehicle No.
   7. License No.
   8. Payment Status.
8. Sort the output file on the basis of amount in ascending order.
9. Send output file to client via email.

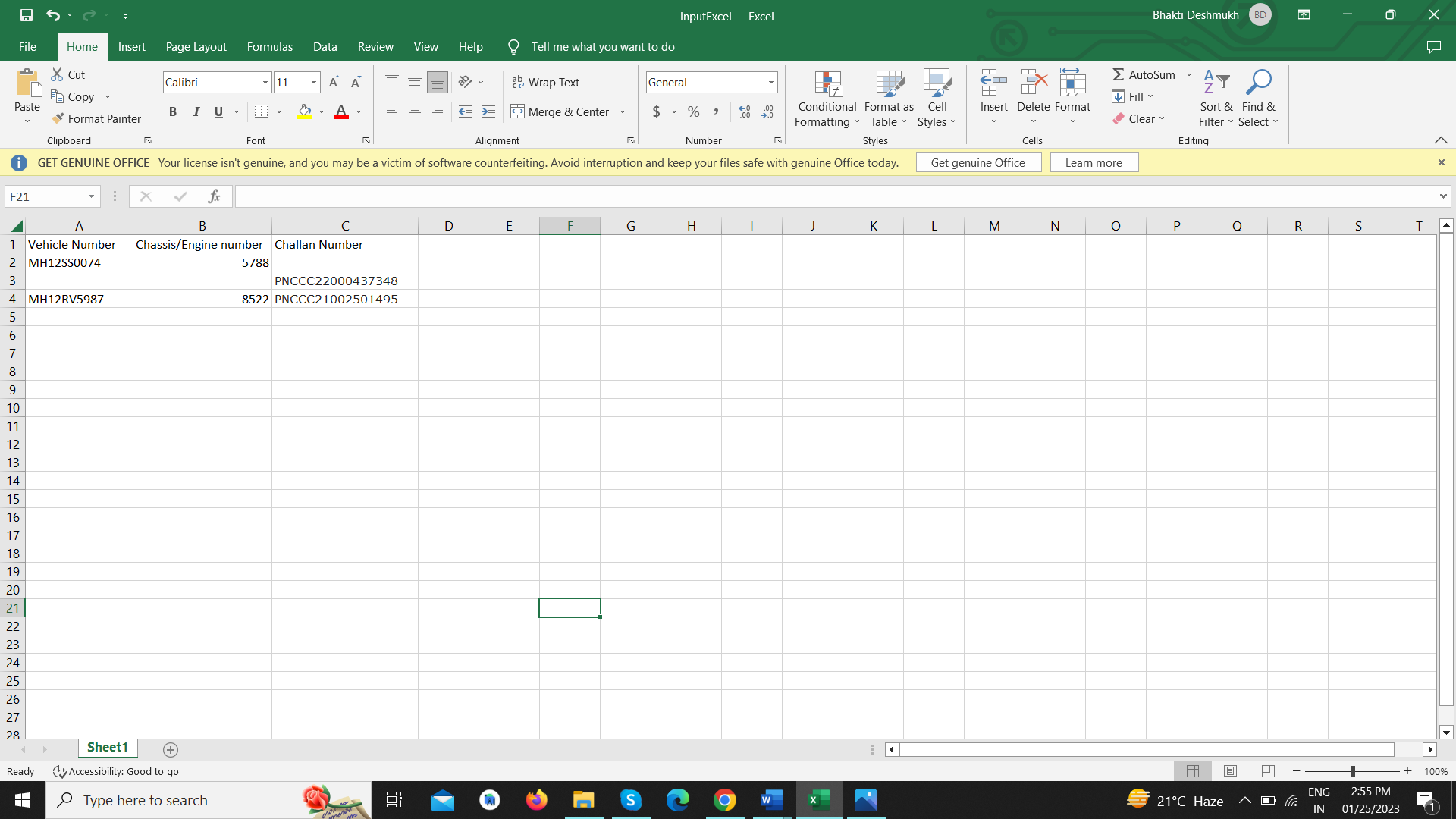
# **Functional Workflow**

## **High-Level Diagram**

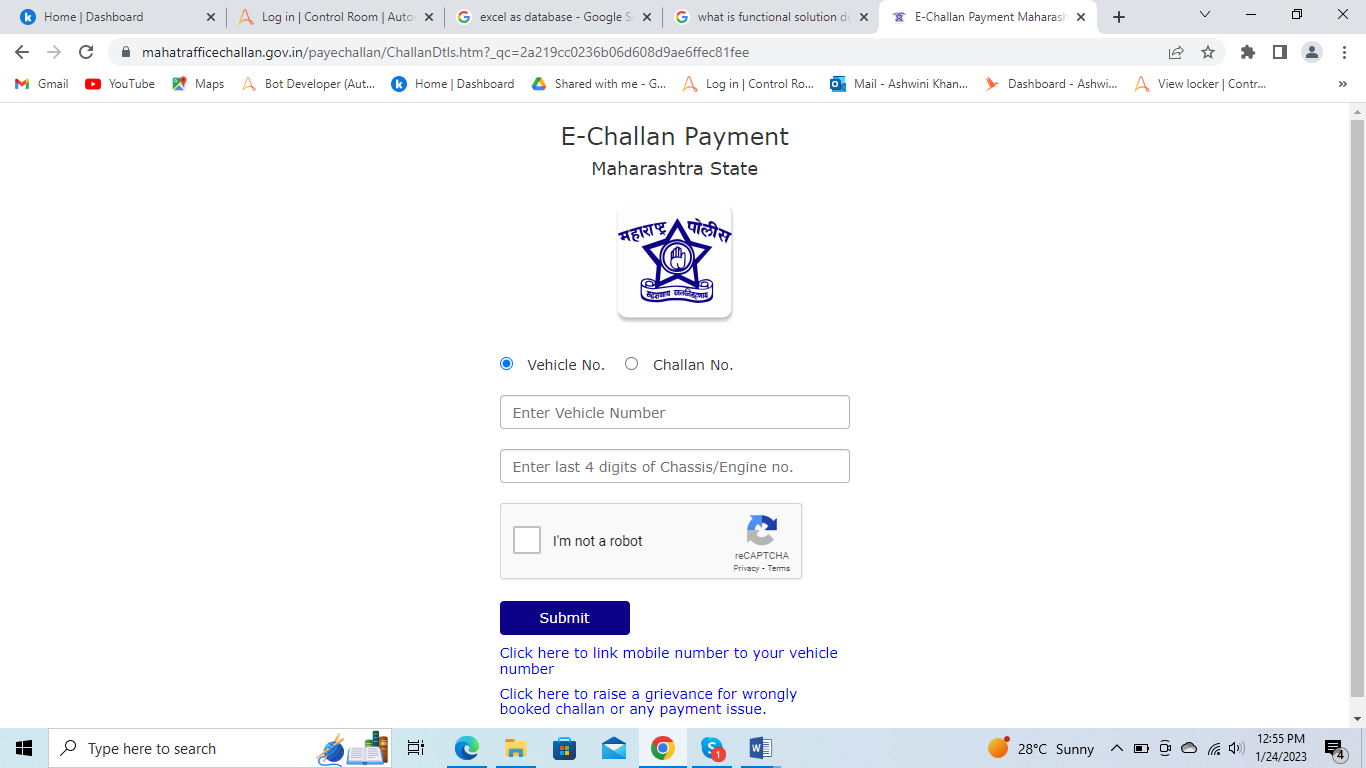
****

## **Detailed Functional Requirements**

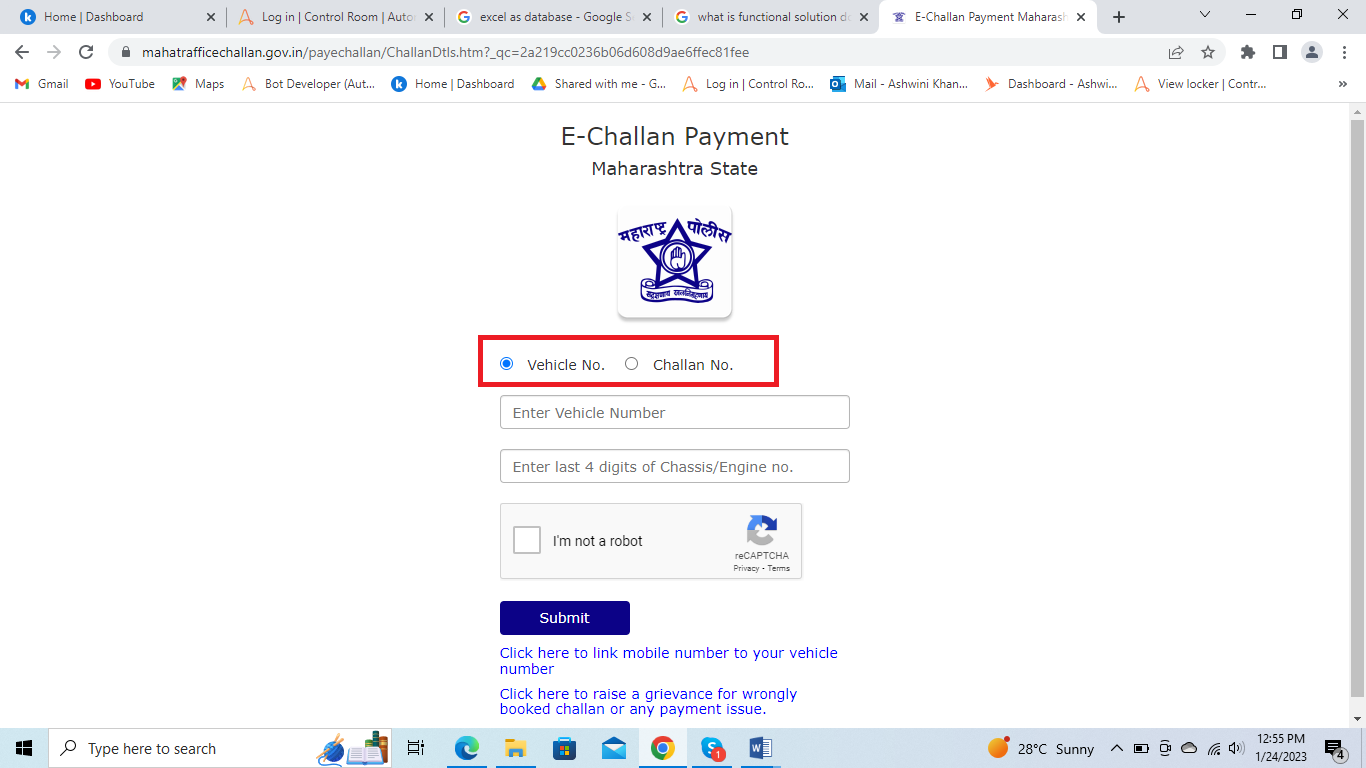
**Step 1:**  Read input file which contain the vehicle number, chassis/engine number as shown in below screenshot.



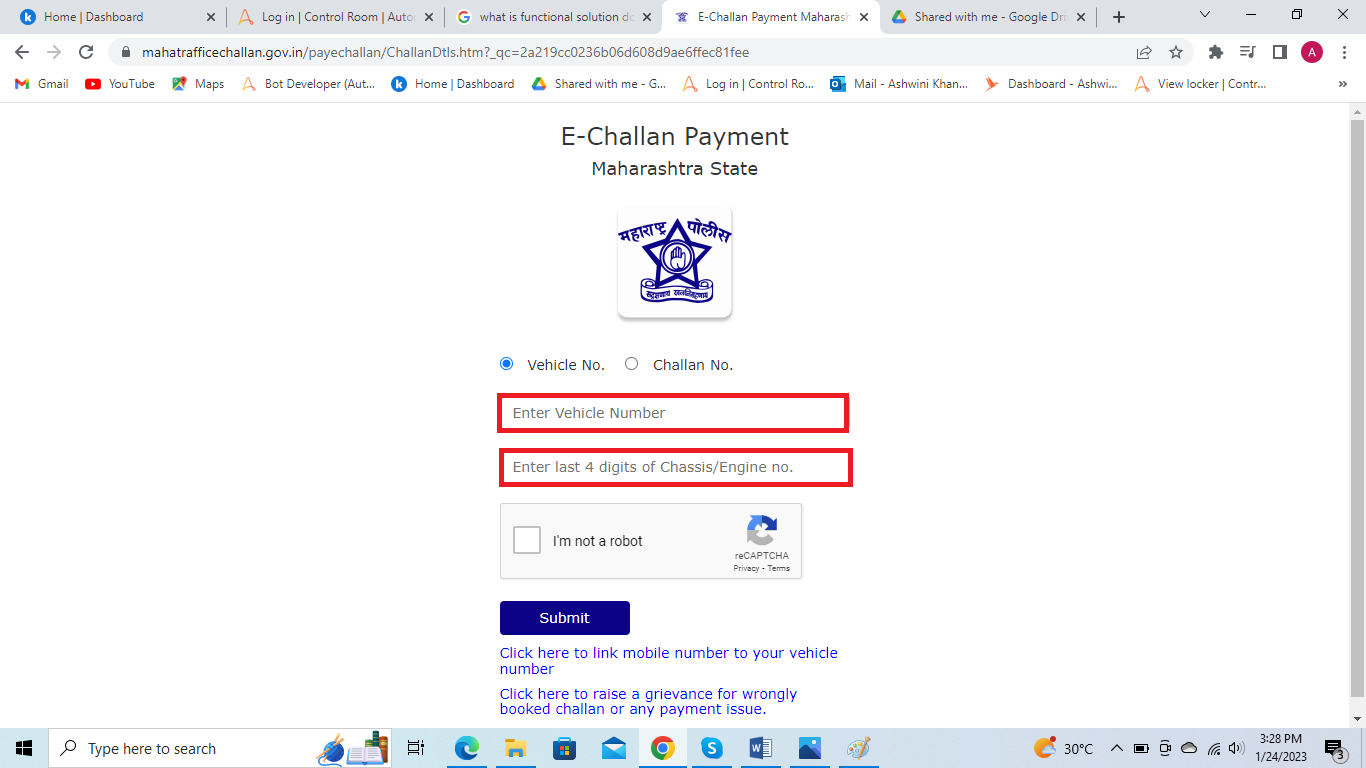
**Step 2:** Open the e-challan payment website through URL: <https://mahatrafficechallan.gov.in/>



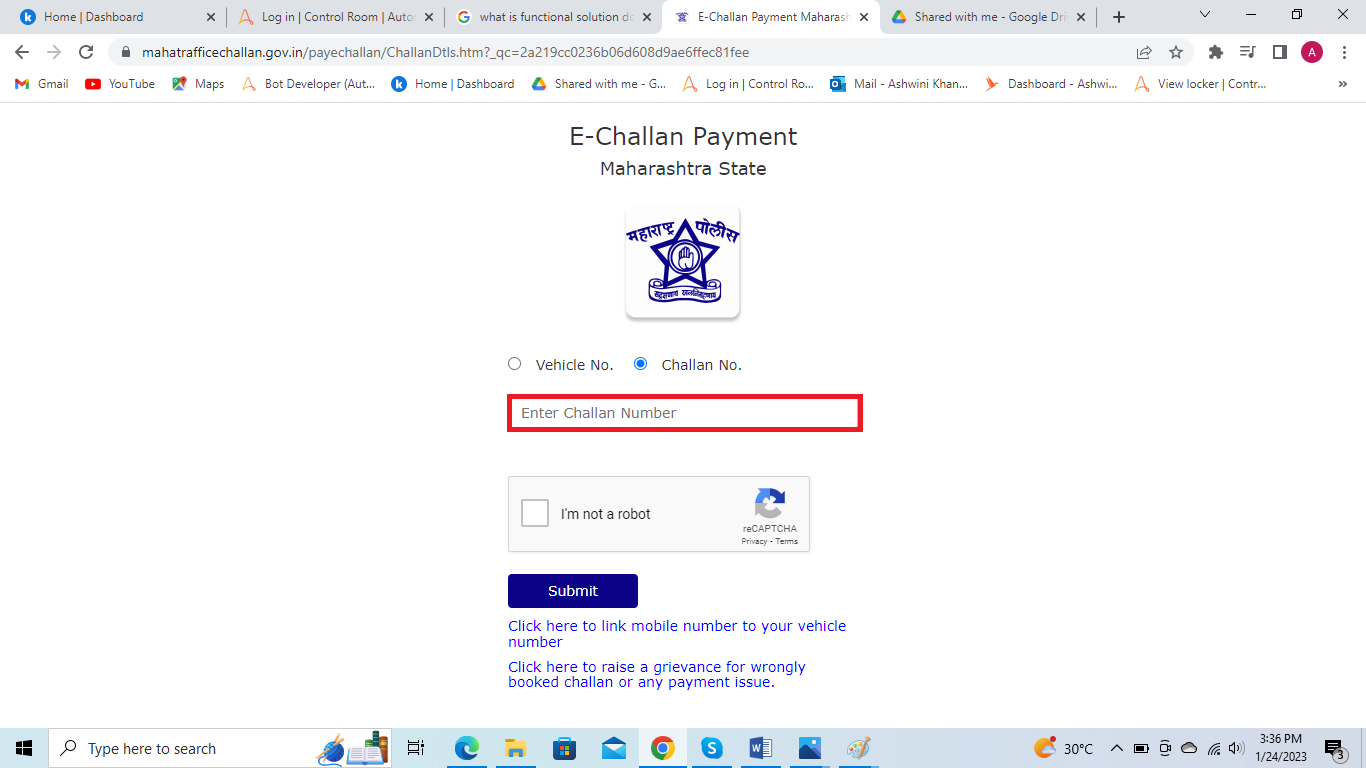
**Step 2:** Select vehicle number or challan number as highlighted in below screenshot.



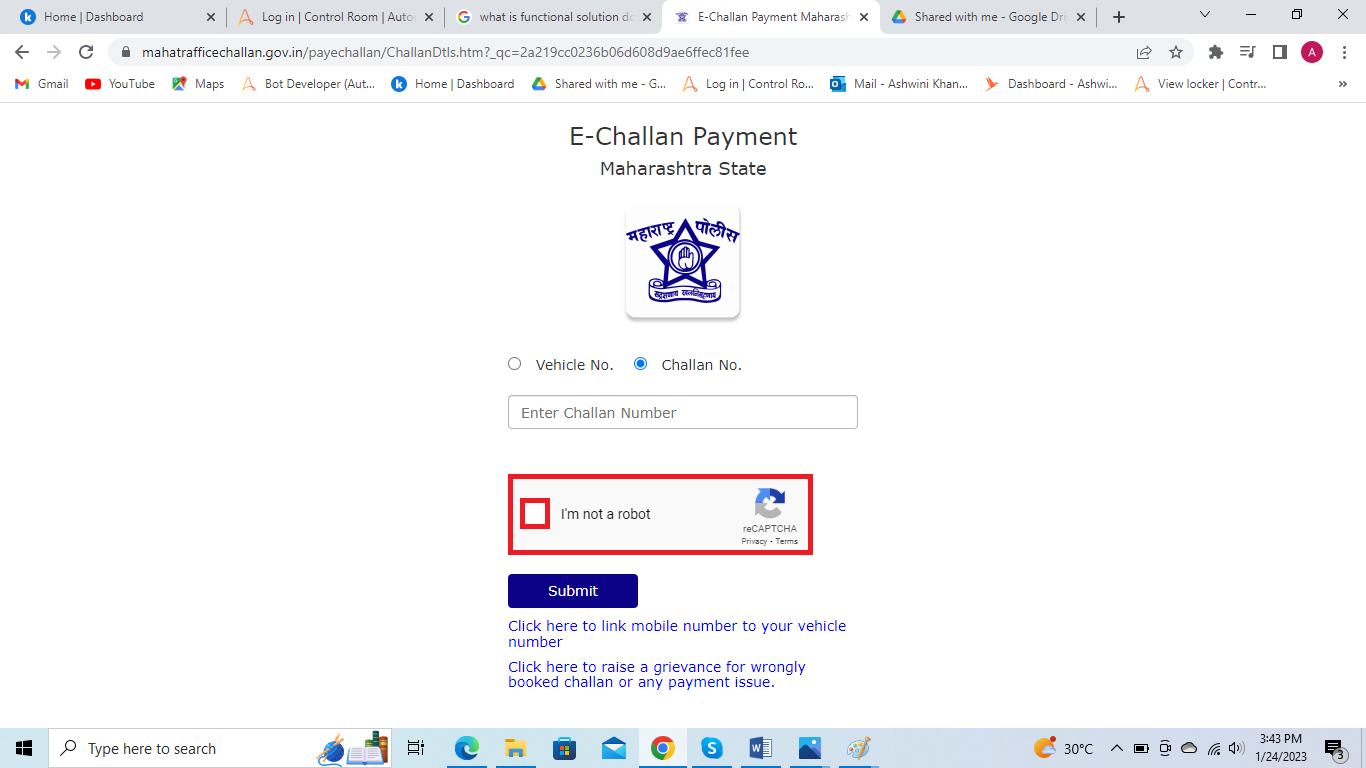
**Step 3:**  If selected vehicle number then enter vehicle number and last 4 digits of chassis/engine number to the fields highlighted in below screenshot.



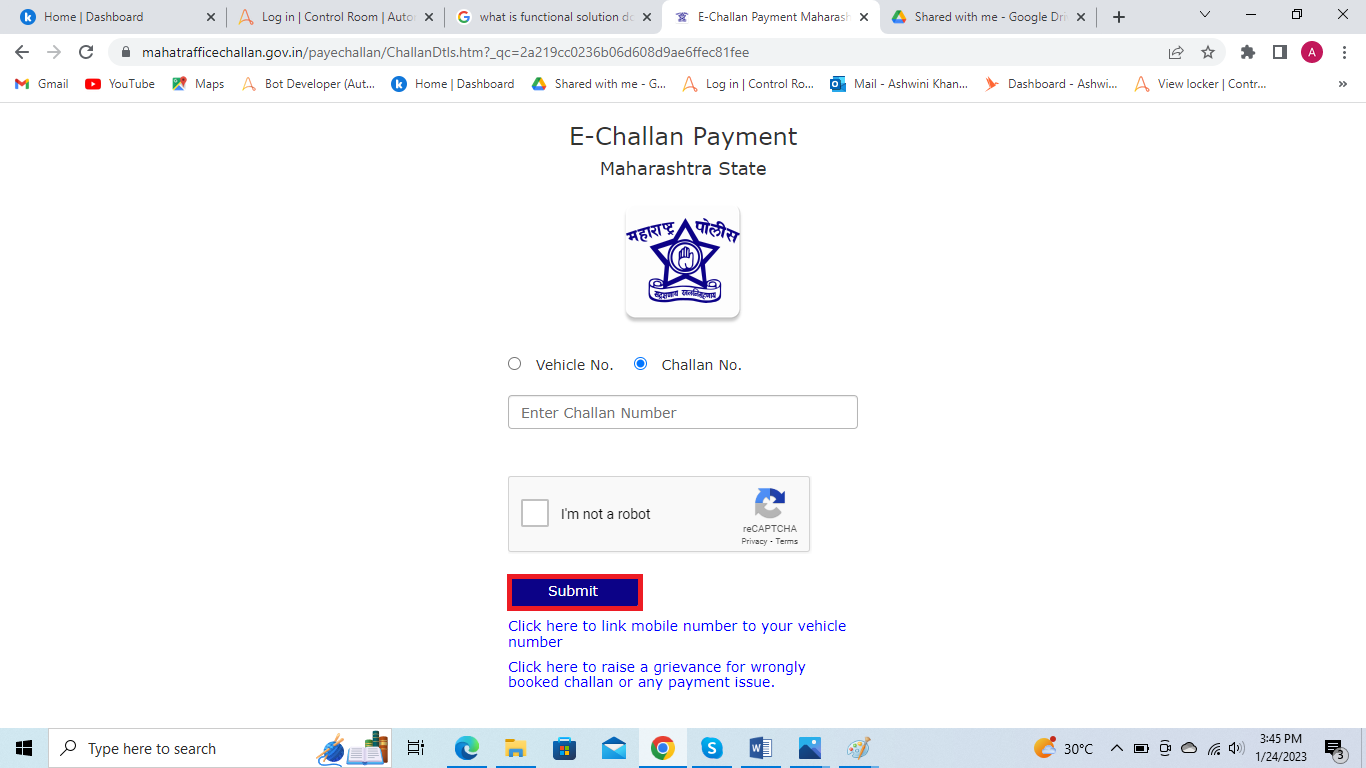
**Step 4:**  If selected challan number then enter challan number in the highlighted field as shown in below screenshot.



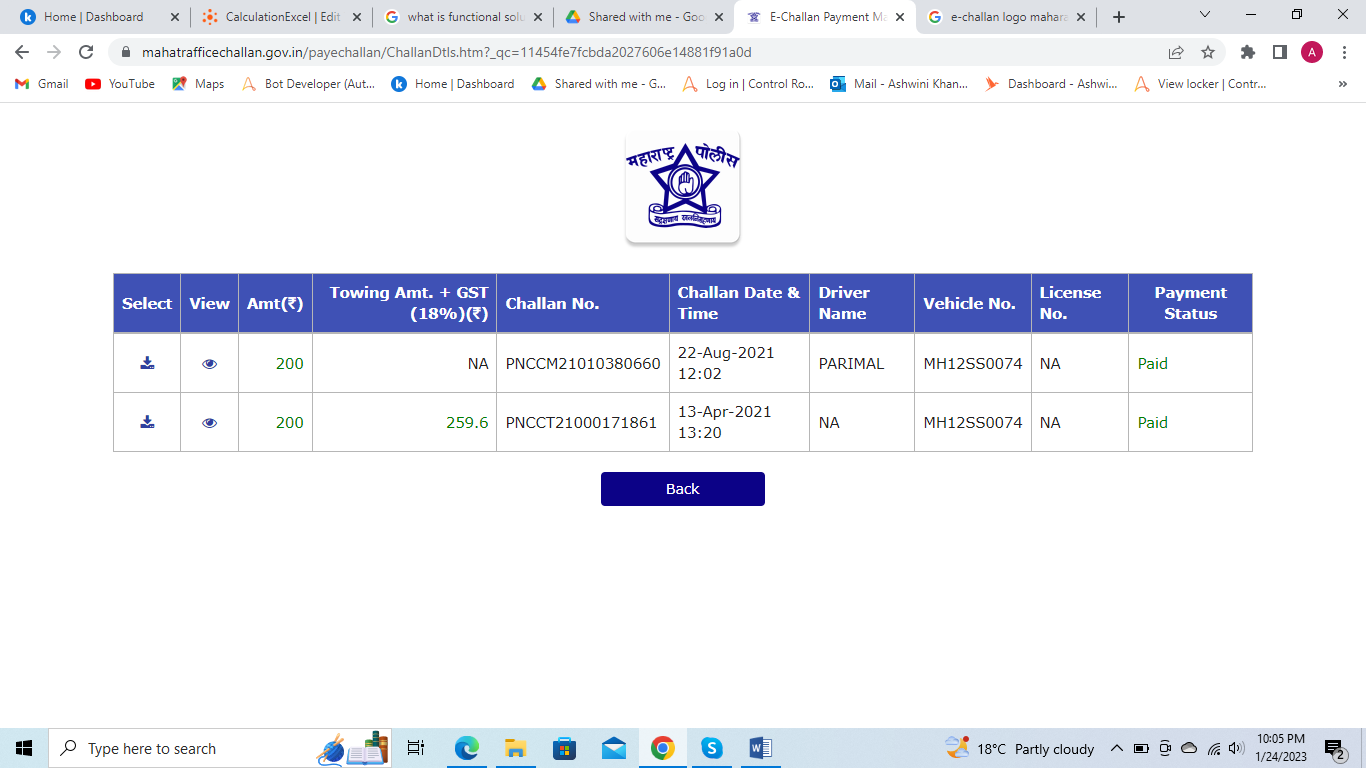
**Step 5.** Click on re-captcha as shown below.



**Step 6:** Click on submit button as shown in the below screenshot.

~~~~

**Step 7:** Challan details page will be loaded as shown below.



**Step 8:** Read below-mentioned challan details and enter it in output file:

8.1 Amount

8.2 Towing Amt. + GST (18%)

8.3 Challan No.

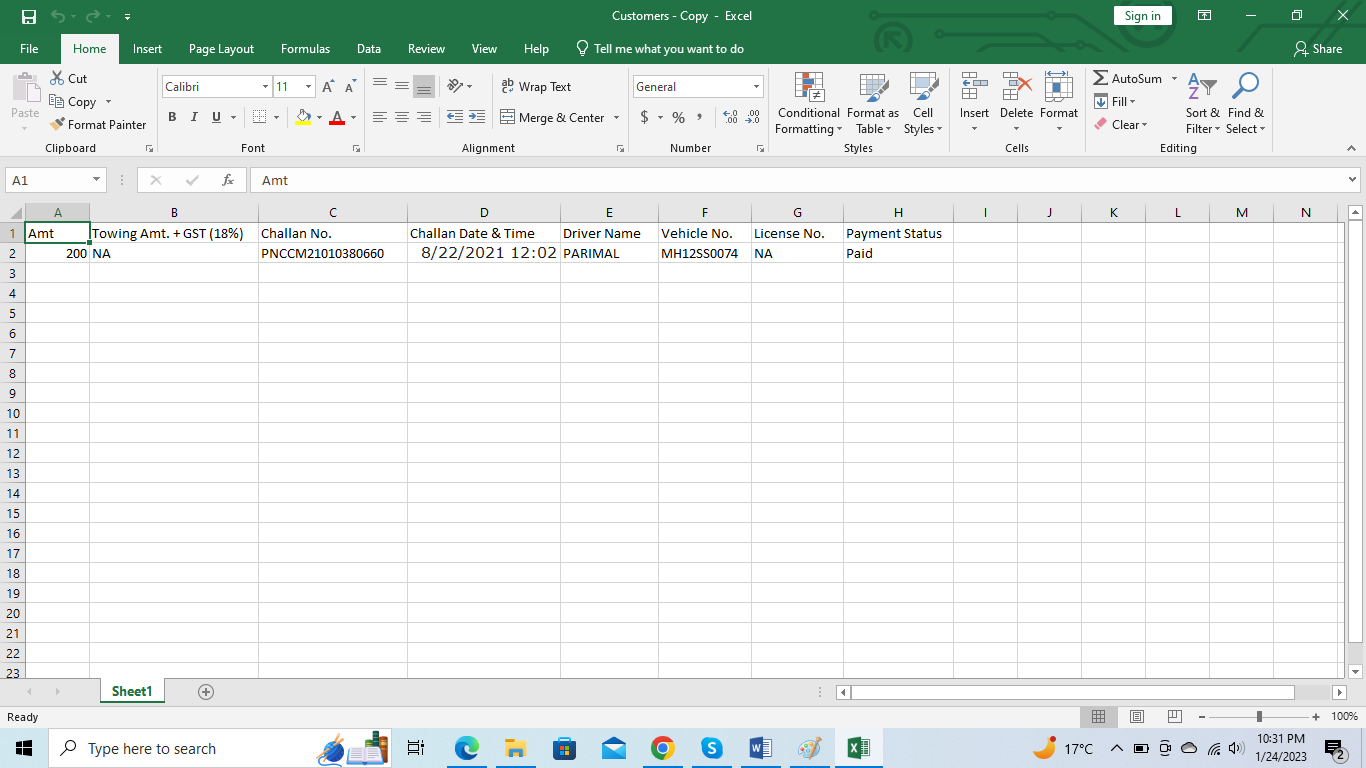
8.4 Challan Date & Time

8.5 Driver Name

8.6 Vehicle No.

8.7 License No.

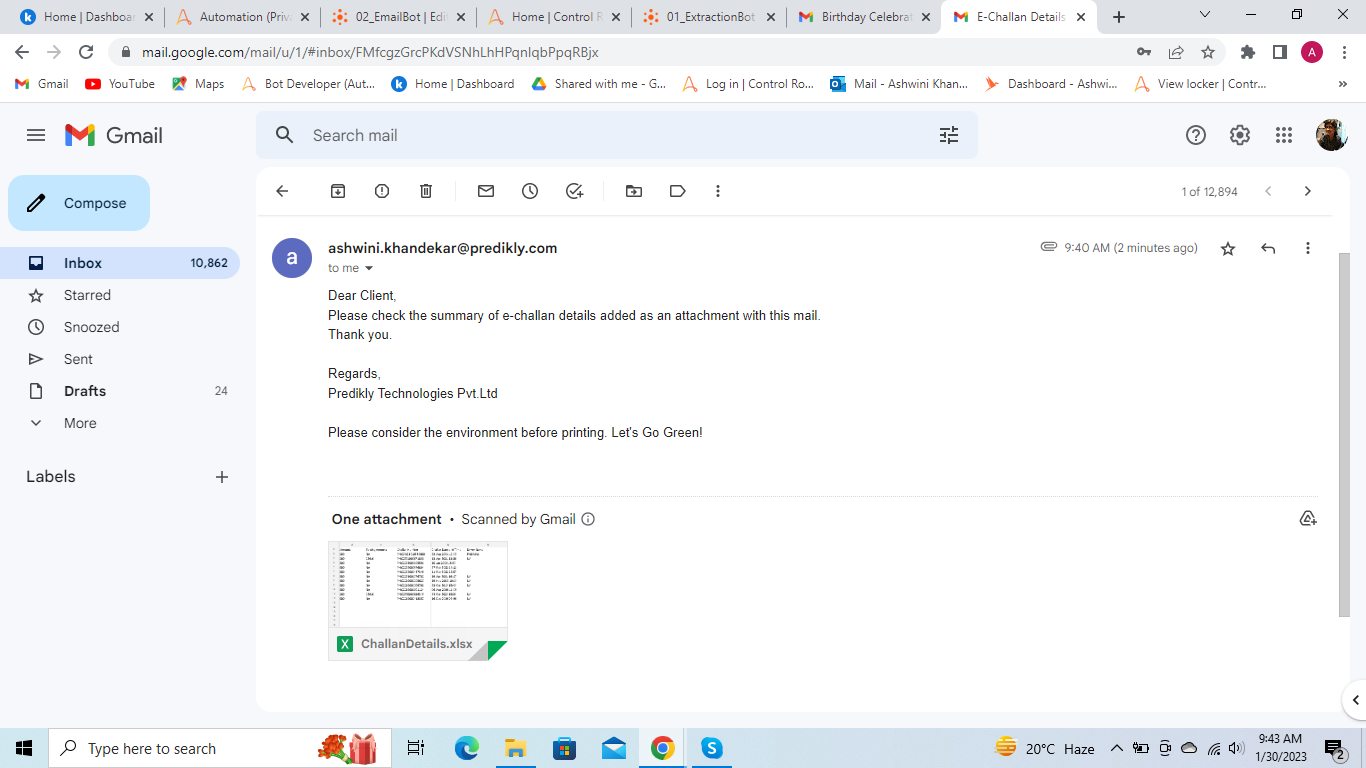
8.8 Payment Status

~~~~

**Step 9:** Repeat step 2 to 8 for each record in input file.

**Step 10:** Sort the challan amount in ascending order and save output file.

**Step 11:** Send output file as an attachment to client via email.



# **Key Assumptions & Dependencies**

The process execution will be **dependent** on following factors:

1. URL of e-challan website (<https://mahatrafficechallan.gov.in/> ) must be accessible.
2. Any changes in the e-challan payment website (<https://mahatrafficechallan.gov.in/> ) will have direct impact on the process.
3. SMTP details must be provided which will be used to send the summary email to the client.
4. Email credentials must be provided to send the summary email to the client.

The following **assumptions** are made while automating the process:

1. Valid input parameters like vehicle number, chassis/engine number and challan number are required to get challan details.
2. For same vehicle, if searched by vehicle number and searched by challan number then it will consider the different records in output file.
3. Folder structure should be there shown as below:
   * + - 1. E-challan Project

1)InputFolder:

i)VehicleDetails.xlsx

2)OutputFolder:

i)ChallanDetails.xlsx

3)ErrorLogFolder:

i)ErrorLogFile.txt

4)AuditLogFolder:

i)AuditLogFile.txt

# **Inputs**

## **Prerequisites**

1. Client email ID must be provided to send the Output file.
2. All necessary environment related access must be granted.
3. All necessary application related access must be granted.
4. MS Excel must be installed.
5. MS Database Engine Redistributable 2010 needs to be installed.

## **Inputs Required**

1. E-challan website URL must be provided by client.
2. Input file with vehicle number, chassis/engine number (last 4 digit), challan number must be provided.
3. Configuration file is needed for accessing the file and web portal.
4. Email ID must be provided for sending an output file as an attachment to the client.

# **Error Handling**

Errors and Exceptions will be handled as follows:

1. Check condition if input file does not exist then stop the process.
2. Check condition if input file does not contain any record then stop the process.
3. In case of incomplete vehicle information, current vehicle will be skipped and next vehicle in iteration will be picked.
4. In case of input file is empty, stop the process.
5. In case of captcha popup exist, stop the process.
6. In case of error while accessing the local machine configuration file, stop the process.
7. In case of e-challan portal error, such as portal down for maintenance etc., stop the process.

# **Out of Scope for RPA**

1. Captcha validation process is not included.
2. Folder structure validation is not included.