**PROJECT**

**ON**

**TESTING - AUTOMATION GUI**

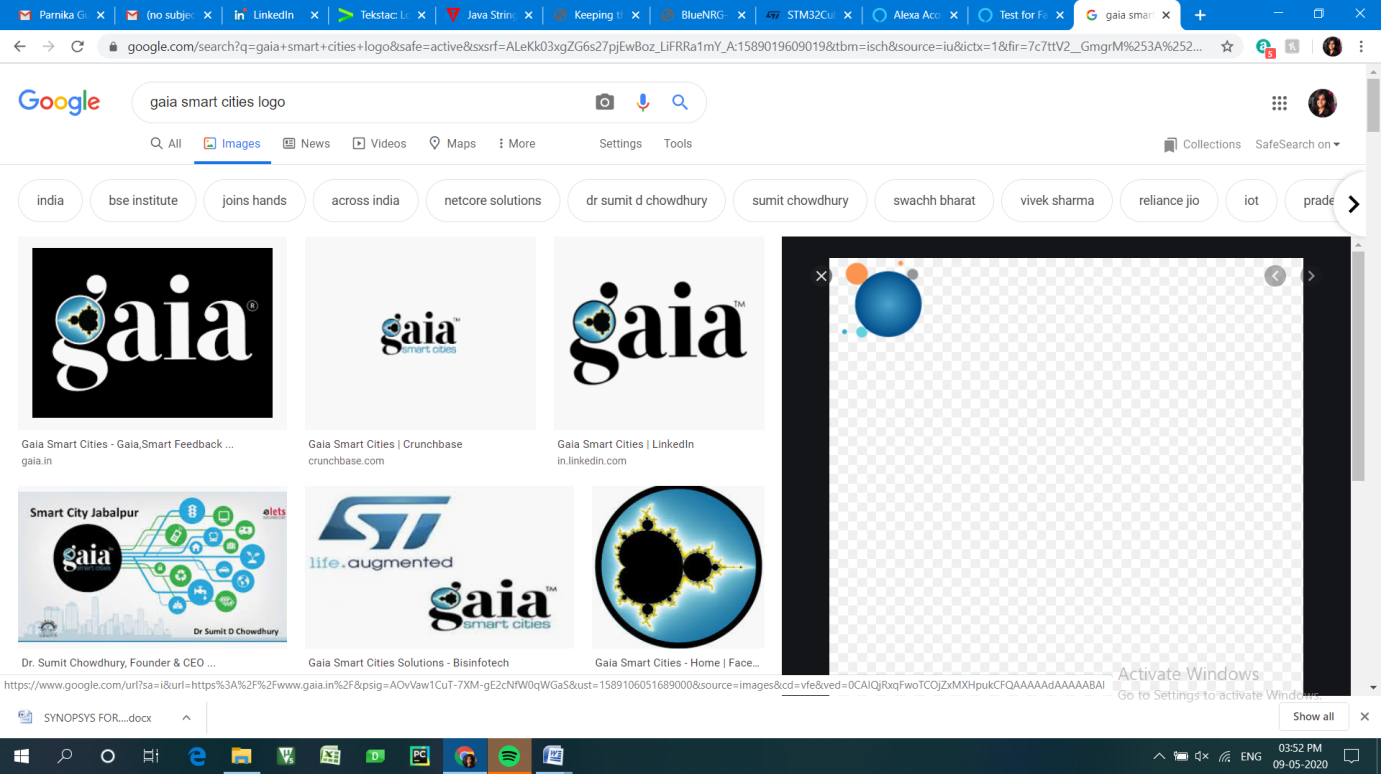
*To be submitted by*

**Parnika Gupta, BTBTE16017**

*Under the supervision of*

**Archit Saraf, Santosh Sharma, Abhishek Behera**

**GAIA SMART CITIES**

****

*For the award of the degree*

*Of*

**B. Tech**

**Electronics & Communication**

****

**DEPARTMENT OF ELECTRONICS**

**BANASTHALI UNIVERSITY**

**BANASTHALI – 304022**

**June 2020**

# Candidate’s Declaration

I hereby declare that the work, presented in the Project, entitled “**Testing-Automation GUI”** in fulfillment for the award of **Degree of “*Bachelor* of Technology”** in Dept. of Electronics with Specialization in Electronics and Communication and submitted to the **Department of Electronics, Banasthali University, Rajasthan** is a record of my own investigations carried under the guidance of Mr. **Archit Saraf, Mr. Santosh Sharma, Mr. Abhishek Behera**, **GAIA Smart Cities**, Greater Noida**.**

I have not submitted the matter presented in this report anywhere for the award of any other Degree.

Parnika Gupta

Student Name: Parnika Gupta

ID: BTBTE16017

Roll No: 1611446

Class: B. Tech (IV)

Branch: Electronics and Communication

College: Banasthali University, Rajasthan

# Table of Contents

[Candidate’s Declaration 2](#_Toc43217072)

[Table of Contents 3](#_Toc43217073)

[List of Figures 5](#_Toc43217074)

[List of Tables 6](#_Toc43217075)

[Abbreviations 7](#_Toc43217076)

[Certificate 8](#_Toc43217077)

[Acknowledgements 9](#_Toc43217078)

[Introduction 10](#_Toc43217079)

[Objective 11](#_Toc43217080)

[Component Description 12](#_Toc43217081)

[Technology used*:* 12](#_Toc43217082)

[Hardware component description: 12](#_Toc43217083)

[Working of Project 17](#_Toc43217084)

[The Project Problem 17](#_Toc43217085)

[Solution Methodology 17](#_Toc43217086)

[App components description 18](#_Toc43217087)

[Interpretation of the output 20](#_Toc43217088)

[Limitation 22](#_Toc43217089)

[Conclusion 23](#_Toc43217090)

[Scope 24](#_Toc43217091)

[References 25](#_Toc43217092)

# List of Figures

[Figure 1: SmartFeedback v7.2 IoT Gateway Board 12](#_Toc43217488)

[Figure 2: GSM Module 13](#_Toc43217489)

[Figure 3: Libraries Used 14](#_Toc43217490)

[Figure 4: Command Flow 17](#_Toc43217491)

[Figure 5: App Components 19](#_Toc43217492)

[Figure 6: GUI- Page 1&2 20](#_Toc43217493)

# List of Tables

[Table 1: Python GUI Libraries 15](#_Toc43217055)

[Table 2: App Components Description 19](#_Toc43217056)

# Abbreviations

* + 1. GUI – Graphical User Interface
    2. PC – Personal Computer
    3. IoT – Internet of Things
    4. v7.2 – Version 7.2
    5. GSM – Global System for Mobile communication
    6. USB – Universal Serial Bus
    7. AT commands – ATtention commands
    8. PCB – Printed Circuit Board
    9. FOTA – Firmware Over The Air

# Certificate

# Acknowledgements

I would like to express my special thanks of gratitude to my teachers and my HOD of Electronics and Communication Dept. in Banasthali University, Rajasthan.

Secondly, I would like appreciate the guidance of my mentors Mr. **Archit Saraf, Mr. Santosh Sharma, Mr. Abhishek Behera in GAIA** who gave me the golden opportunity to do this wonderful project on the topic “**Testing-Automation GUI**”, that also helped me to connect with recent technologies and I came to know about so many new things, I am really thankful to them for their collaboration during the entire internship period.

Thirdly, I would also like to thank my parents and friends who have been so supportive while completing the project within the limited time frame.

# Introduction

The GAIA Smart Cities is a company working in the field of IoT and artificial intelligence. Its core PCB board is SmartFeedback v7.2 IoT Gateway board, designed and managed in India. Since the product started rolling out more and more in the market, there was a crucial requirement to shift the testing of the product from manual to automatic testing.

This project is mainly aimed to automate the whole testing process for the main board of the company by verifying and giving a report of the testing parameters through a GUI.

# Objective

The project aims to configure and test the SmartFeedback v7.2 IoT gateway board through python automated GUI script remotely, by using GSM module (compatible with 4G and 2G both sim cards).

The end product is an automated GUI, instructing the GSM module attached with the PC by USB serial port, thus creating the communication between the GSM module and the SmartFeedback board.

# Component Description

## Technology used*:*

* **Hardware-**
* ***SmartFeedback board*** – 7.2 version IoT gateway PCB board by GAIA
* ***GSM Module*** - USB to GSM, direct connection to PC that acts as an intermediate between GUI and SmartFeedback v7.2 board
* ***4G / 2G Antennas*** – Attached to both GSM module and SmartFeedback v7.2 board
* ***4G / 2G SIM cards*** – Inserted in both GSM module and SmartFeedback v7.2 board SIM slots
* **Software-**
* ***Pycharm*** – PyQT5, Serial, Pandas python libraries for desktop app development
* ***Qt Designer*** – For generating QT app view
* ***TeraTerm*** – For initial interface testing

## Hardware component description:

1. ***SmartFeedback board***

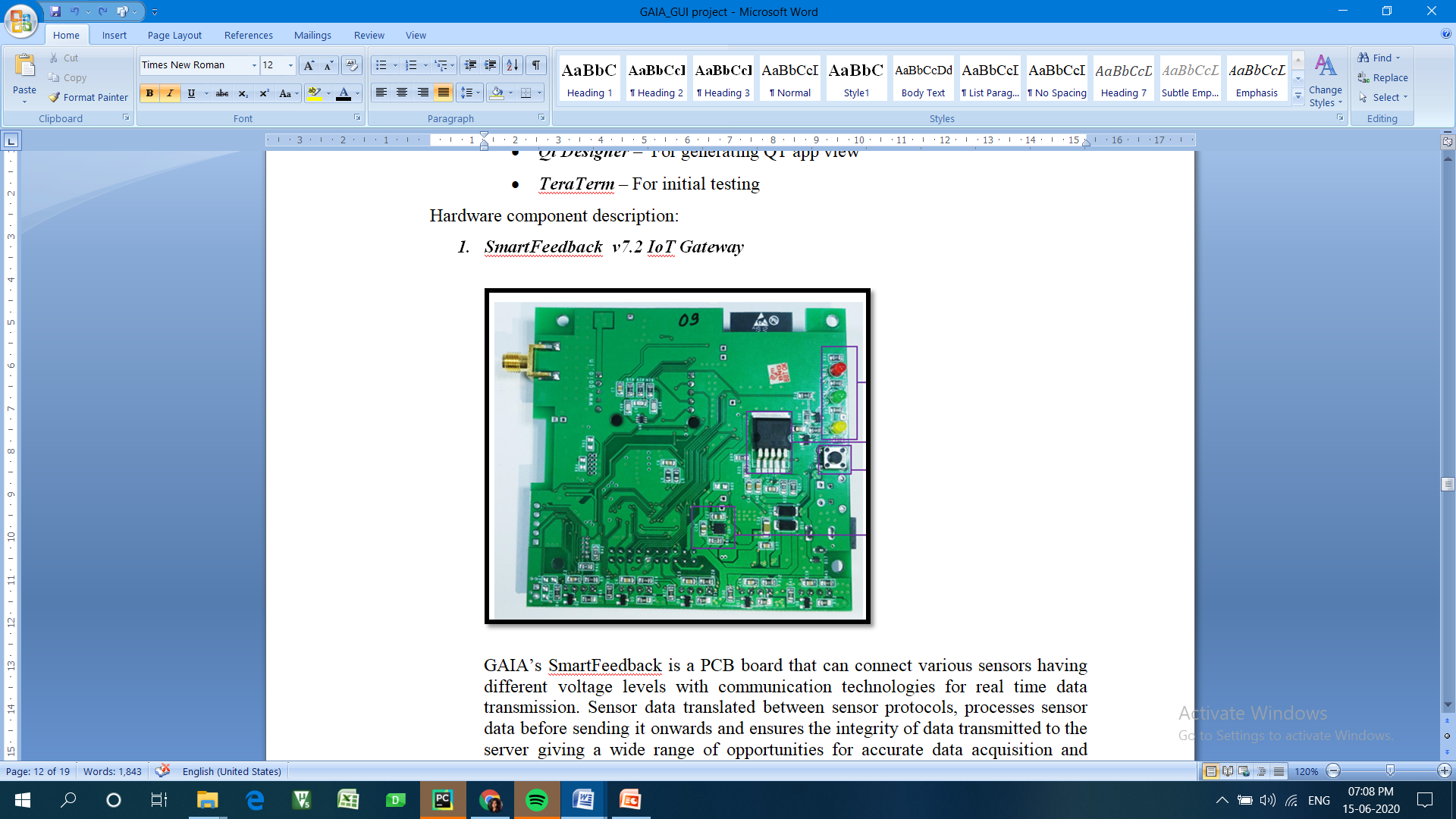
******

Figure 1: SmartFeedback v7.2 IoT Gateway Board

GAIA’s SmartFeedback is a PCB board that can connect various sensors having different voltage levels with communication technologies for real time data transmission. Sensor data translated between sensor protocols, processes sensor data before sending it onwards and ensures the integrity of data transmitted to the server giving a wide range of opportunities for accurate data acquisition and processing at the edge.

SmartFeedback IoT nodes perform following functions:

* ***Multi sensor and protocol bridge***- Connects different sensors, translates and transfers data among or between systems with different communications protocols (digital, analog, RS232, RS485, USART, I2C, and SPI etc).
* ***Daughter Board slot***- Connector of 20 pin, have plug in multiple sensors (up to 12 sensors interfacing over protocols) or communication modules
* ***Trusted connectivity and security***- Ensuring the integrity of the network and system in both directions
* ***Storage and analysis***- Onboard intelligent firmware coupled with onboard storage drives processing, intelligence and decision making closer to the device
* ***Upgradable***- By Firmware Over The Air (FOTA)
* ***Connectivity*-** GSM, Wi-Fi and BLE connectivity on-board, GSM communication is used in this project

1. ***GSM Module***

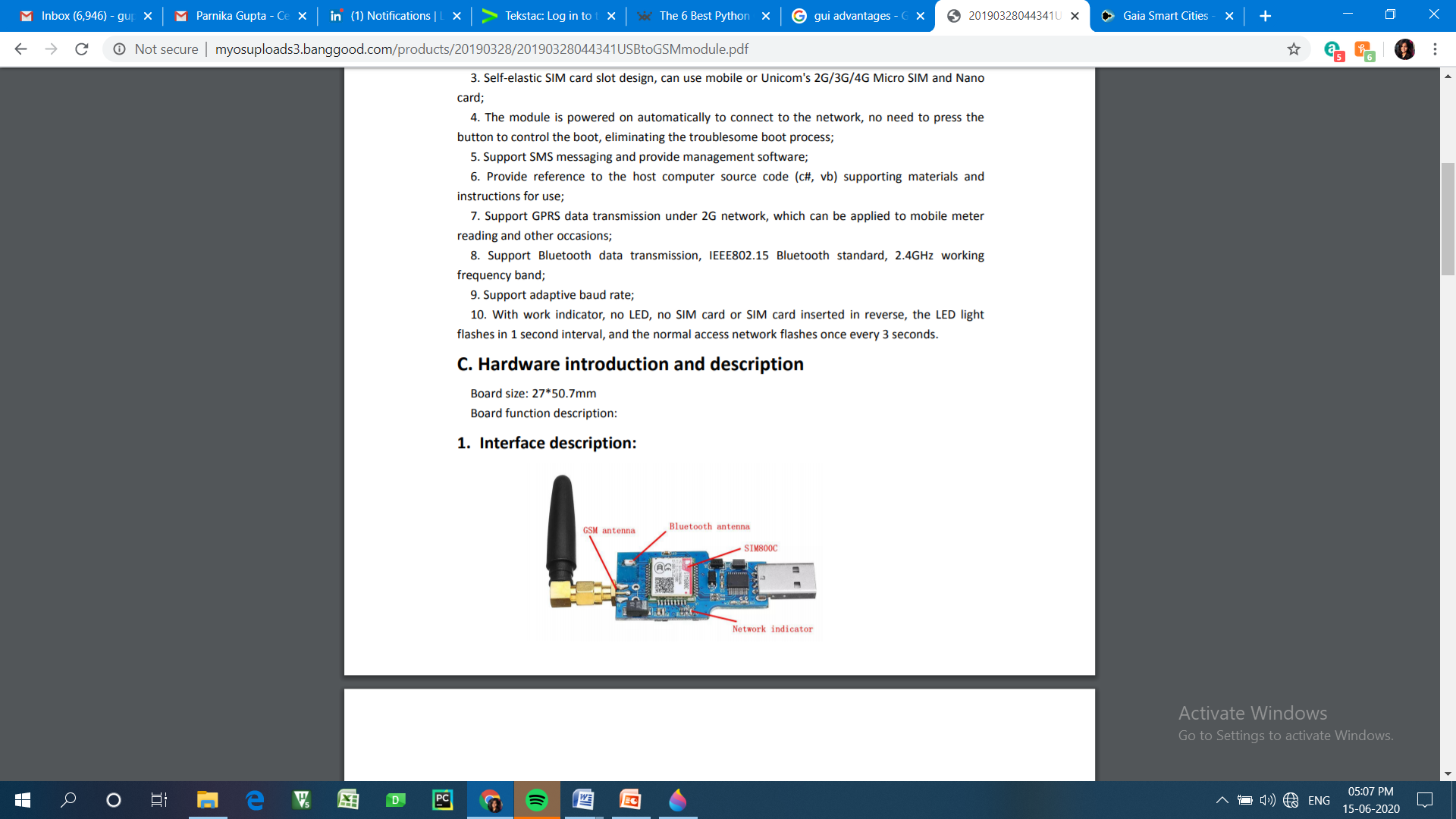
****

Figure 2: GSM Module

GSM module is a USB to GSM quad-band GSM/GPRS module with stable performance, compact appearance and high cost performance contains SIM800C operating at GSM/GPRS 850/900/1800/1900MHz standard. It can also transmit voice, SMS information and data information with low power consumption, and can be applied to various compact product design requirements.

**Instructions for use**:

1. 2.4GHz GSM antenna connected to the module and the SIM card inserted.
2. Plug the GSM module in USB port of the computer.
3. After starting module, the indicator light flashes once in 1 second. After a period of initialization, the indicator light changes to 3 seconds confirming that it is in the network. Now, SIM800C GSM module can be controlled with the AT command.
4. ***Note***:
5. Micro/Nano card are supported but does not support 3G/4G network and telecom card
6. When using for the first time, the serial port driver of CH340 is mandatory to install.
7. ***GUI***

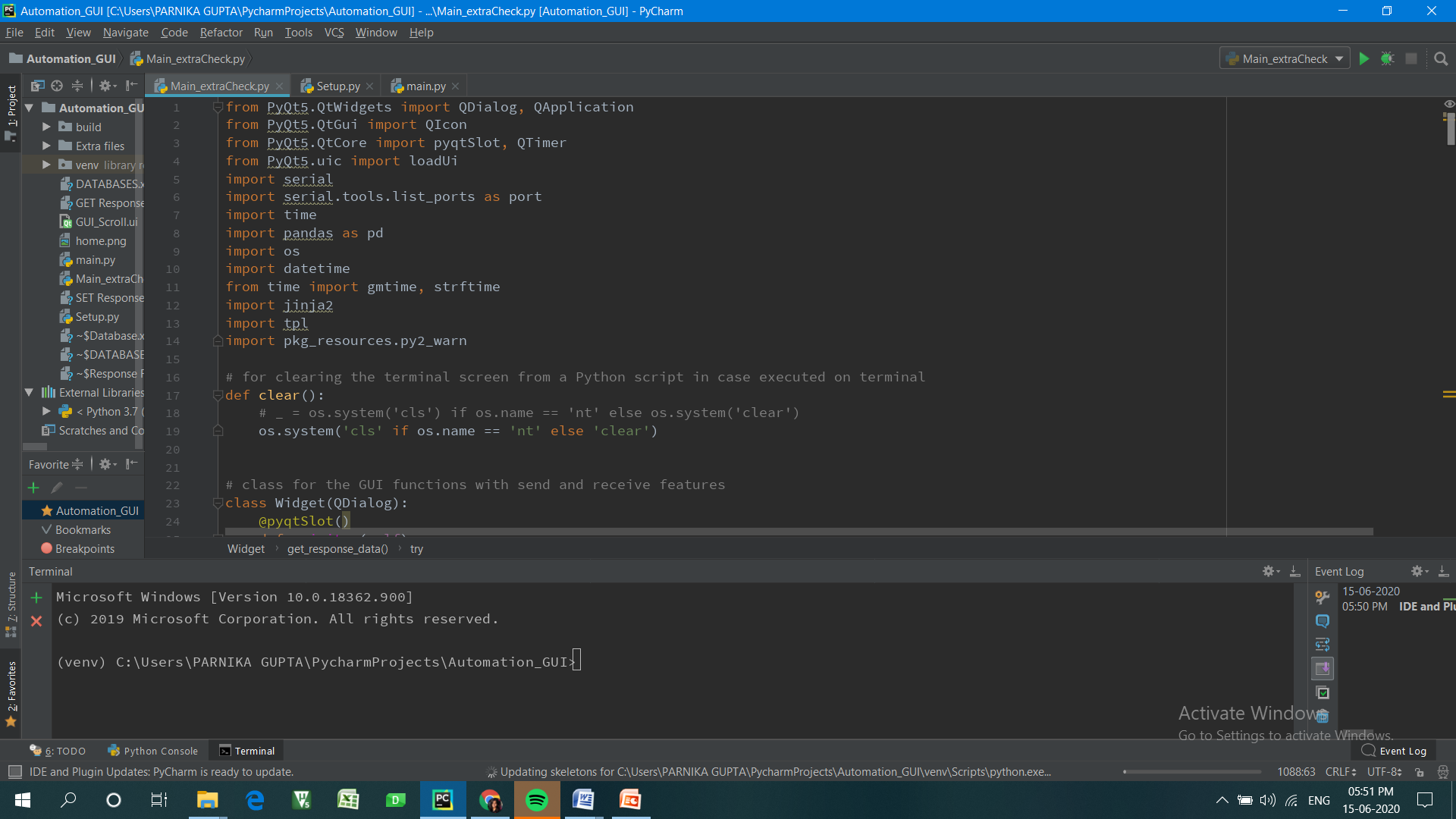
******

Figure 3: Libraries Used

Table 1: Python GUI Libraries

|  |  |  |
| --- | --- | --- |
| **Python Libraries** | **Description** | **Supported by** |
| **Kivy** | An open source OpenGL framework for creating new user interfaces having toolkit of 20+ widgets | Raspberry Pi  MacOS  Android iOS  Unix/Linux  Windows |
| **PyQT** | Favored Nokia cross-platform implementing the Qt library for the Qt application development framework using Qt Designer to create visual dialogs | Unix/Linux  Windows  Mac OS X  Sharp Zaurus |
| **Tkinter** | It is old, open source and use Tk, popular for its simplicity and graphical user interface with abundant resources, codes and reference books | Unix/Linux  Windows  Mac **OS** X |
| **WxPython** | Open source cross-platform wrapper GUI library WxWidgets implemented as a Python extension module | Windows  Mac OS  Unix/Linux |
| **PyGUI** | Graphical cross-platform application framework, simplest and lightweight of them all, as the API is purely in sync with Python | Unix/Linux  Macintosh  Windows |
| **PySide** | Free cross-platform UI framework, GUI Qt toolkit by Nokia. Contains network, multimedia, XML documents, databases, GUI, API compatibility with PyQt4 | Linux/X11  Mac OS X  Maemo  Windows  Android in future |

**Reason to choose PyQT5 for GUI out of the above:**

1. **Compatible with Serial library of python**
2. **Has its own module for time (QTimer) for delay with extended features**
3. **Compatible with most of the operating systems**
4. **Moderate complexity**

# Working of Project

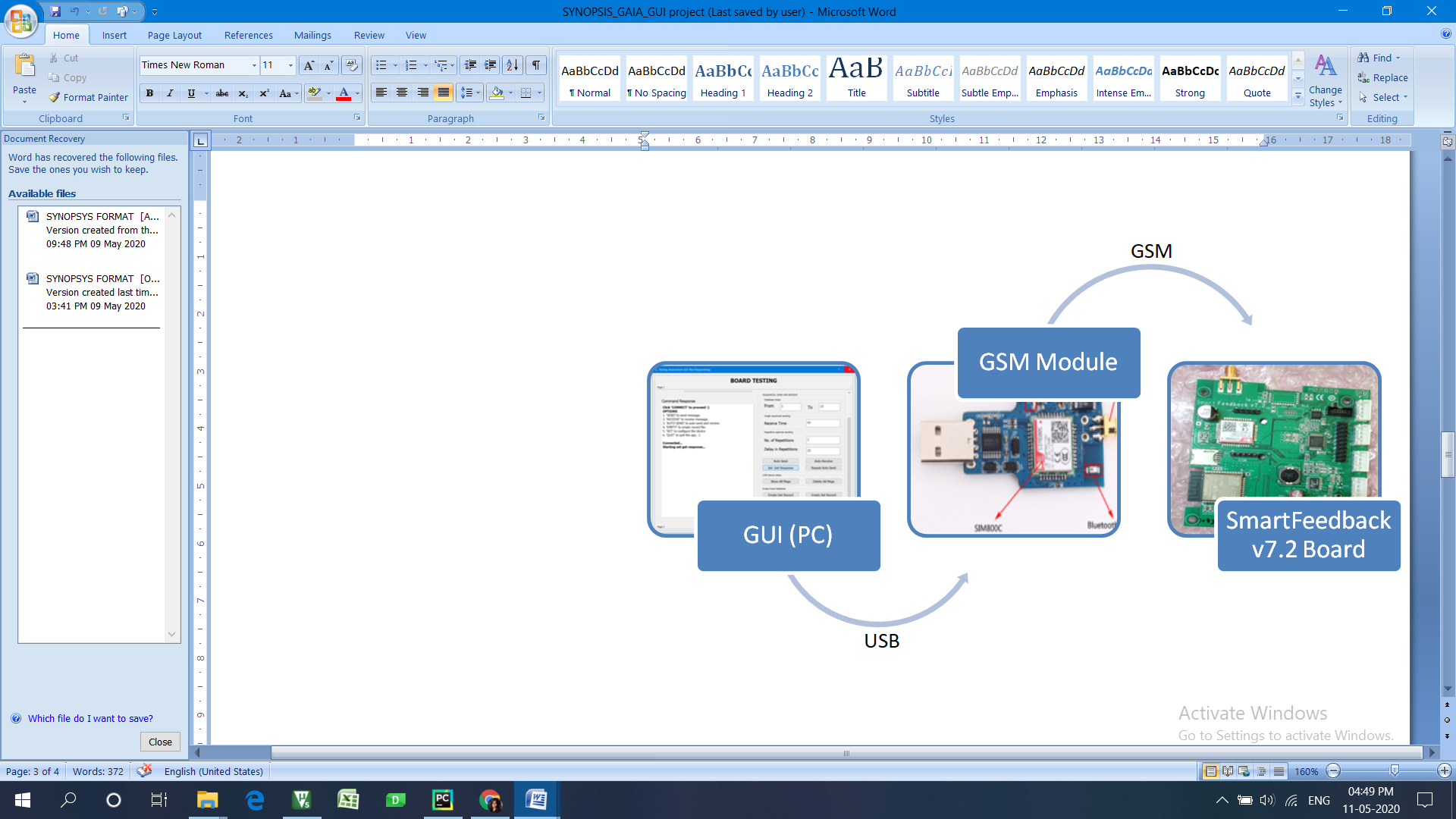
****

Figure 4: Command Flow

## The Project Problem

Requirement for the overall project is as follows:

1. Convert manual testing to automatic testing
2. Proper record generation in the form of excel sheets
3. Configuring 1000s of on-field gateways with 40+ SMS at a time
4. Save time and productivity
5. Reduce chances of error than in traditional testing
6. Reports can be verified automatically
7. Product should be user-friendly

## Solution Methodology

***Step1****-* Figure out the hardware components needed for making the project

***Step2***- Assemble them by checking the technical feasibility, interface testing:

* *to check the communication between the GSM module and PC*, AT commands sent by TeraTerm (HyperTerminal software), that indicated the script will able to automate via python GUI libraries
* *to check the communication between the GSM module and SmartFeedback board*, AT commands on TeraTerm are used to send messages to SmartFeedback board

***Step2****-* Gather the SET and GET SMS command list and AT commands list to be used, from reference documents

***Step3****-* Checking SMS sending from GUI to SmartFeedback board using the command list from the lookup table

***Step4***- The testing/configuration while sending SMS from script to SmartFeedback board procedure:

* Sequential sending and getting SET as well as GET SMS commands through an editable look-up table
* SET SMS check- SUCCESS, FAILURE or NO RESPONSE of the incoming SMS data into report
* GET SMS check- Verification of the parameters in the report

***Step5****-* Make continuous updates to improve the automation script to accommodate more features

**Step6**- Prepare documentation

## App components description

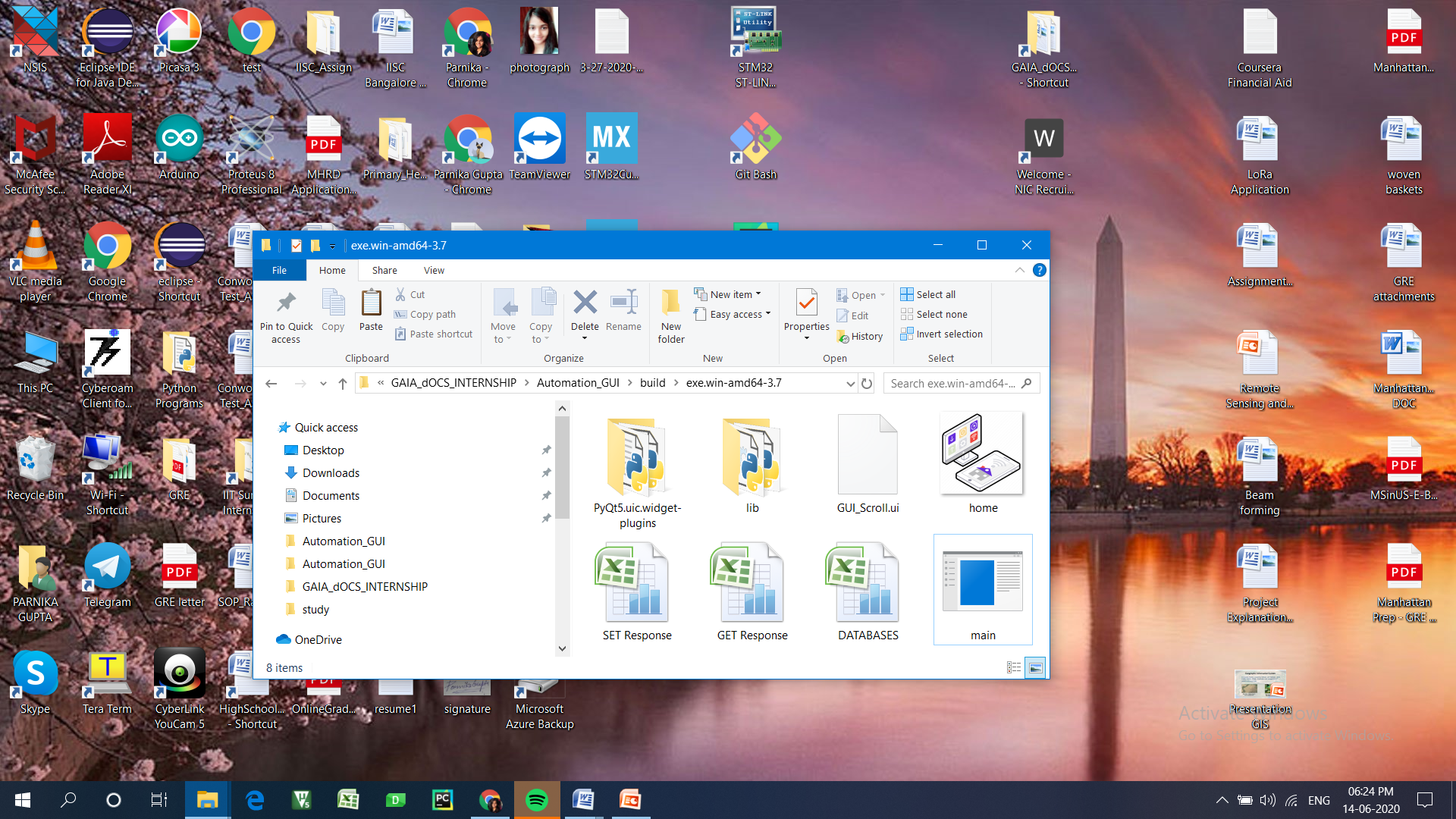
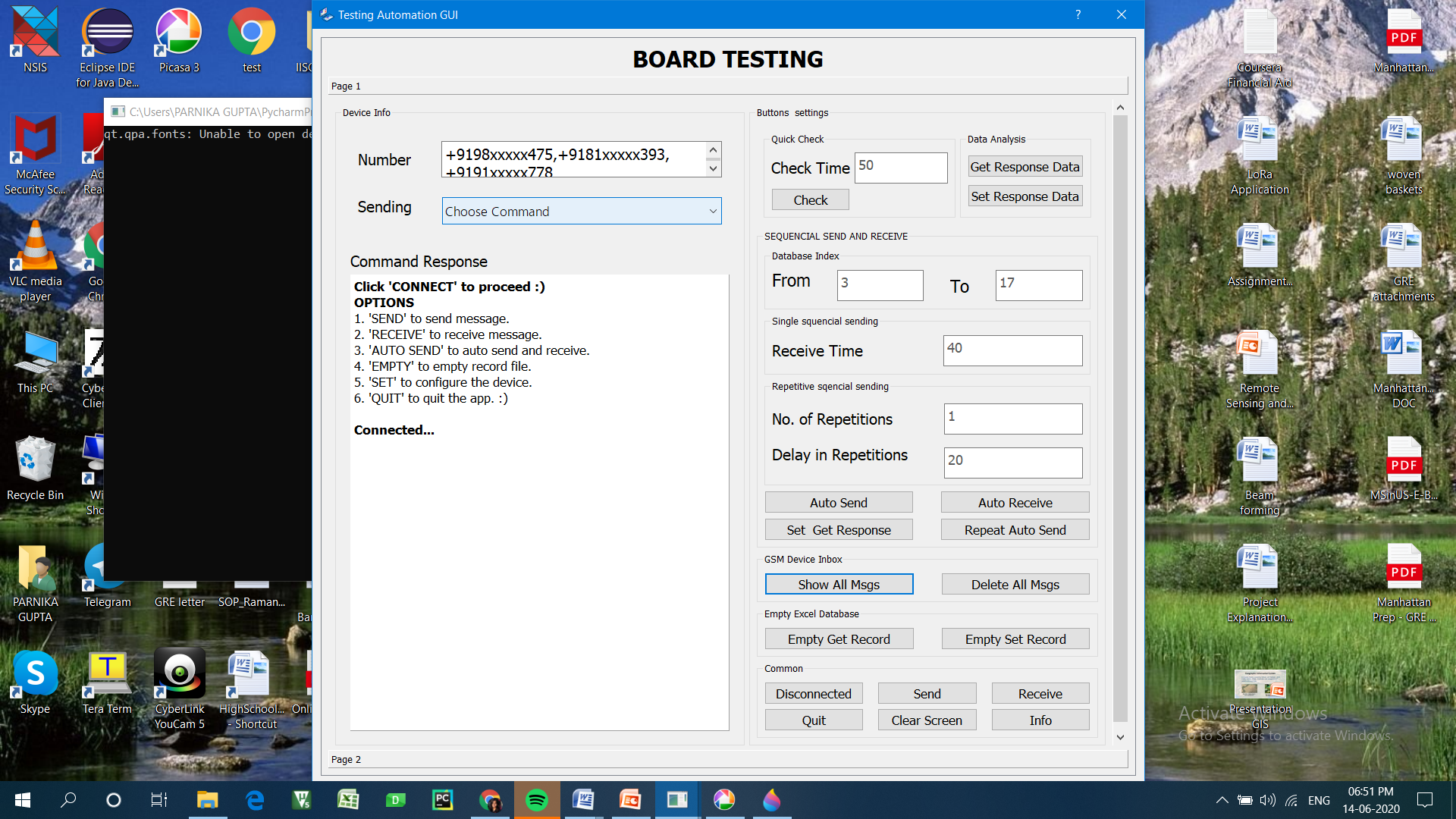


Figure 5: App Components

Table 2: App Components Description

|  |  |  |  |
| --- | --- | --- | --- |
| **File Name** | **File Description** | **Software** | **Package Used** |
| **Lib, Plugins** | Supportive python files generated while making the app independent | NSIS app | CX-freeze |
| **Main** | Main Application | PyCharm Editor | PyQt5 & Serial |
| **Database** | Editable lookup file | MS Excel | Pandas |
| **GET & SET Response** | Verifiable reports | MS Excel | Pandas |
| **GUI\_Scroll** | App view designing | QT Designer | PyQt5 (loadui module) |
| **Home** | App Icon | Jpeg file | - |

## Interpretation of the output



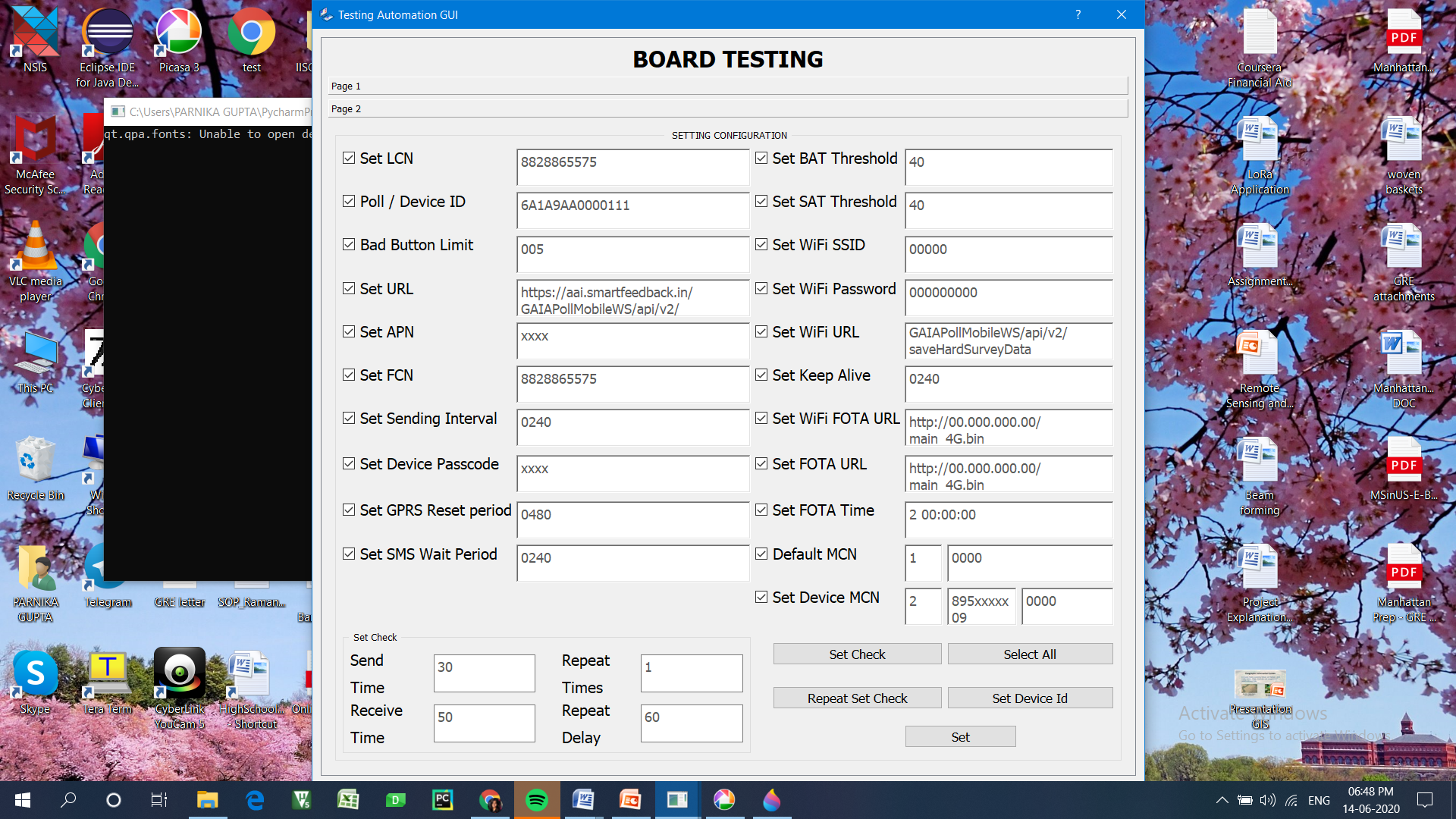


Figure : GUI- Page 1&2

Interpretation of the output is determined by the performance of the GUI, which is created to do the following operations:

1. Sequentially send and receive the SMS commands to the on-field SmartFeedback boards, provided both devices are in network, in minimum time
2. Able to configure and test multiple SmartFeeback boards automatically
3. Sequence of command list for configuring as well as testing SET and GET commands should be interchangeable and the delay sending them is editable by the lookup table
4. Test is able to repeat on user’s will
5. Able to handle exceptions
6. Able to generate a report of failure or success of the SET messages and verification of GET messages with respect to the time of test and the device contact number
7. Guides the user while changing parameters and handling exceptions
8. Script is a desktop app, compatible to run by a setup on PC irrespective of its dimension

# Limitation

* *Connection issue*- Good GSM network needed all the time
* *For few messages mobile takes lesser time*- Sending the configuration commands sequentially to many SmartFeedback board takes a longer time since first it sends and then collects the response messages from the GSM module, therefore slow
* *Dependency of script on hardware*- Other than GSM connection issues, the hardware can once in a while become faulty, or there can be network loss, which is a problem for the script.

# Conclusion

The automatic testing is very useful in the long run. A user-friendly platform benefits the user to operate testing on click of a button and the whole SmartFeedback v7.2 IoT gateway board is configured, tested, analyzed and evaluated which otherwise requires constant hectic message sending via phone and manual checking of the received message responses from the gateways.

# Scope

The scope of the project is determined by the following:

1. *Profitable* –

Without an automation script, several testers are required, which increases the monthly costs of the company. With this script, the testing work can be performed by only one or two people, which is huge savings in the long run, increasing the productivity of the employees.

1. *Helpful* –

Project very helpful since testing is a redundant task and it’s good to have a proper report at the end, that may not be possible if the 40+ messages has to be sent for one round of test and if we have to repeat the task for configuring 1000s of on-field gateways then it is a problem therefore the GUI will be helpful to the testers as well as to the on-field teams, hence saving time and using their productivity in other tasks making the GUI profitable to the company

# References

* *GSM Module link*:

http://myosuploads3.banggood.com/products/20190328/20190328044341USBtoGSMmodule.pdf

* *SIM7600E AT commands link (In SmartFeedback board)*:

<https://simcom.ee/documents/SIM7600C/SIM7500_SIM7600%20Series_AT%20Command%20Manual_V1.01.pdf>

* *SIM800C AT commands link (In GSM Module)*:

<https://www.elecrow.com/wiki/images/2/20/SIM800_Series_AT_Command_Manual_V1.09.pdf>

* *Python PyQt5 library link*:

<https://doc.qt.io/qtforpython/>