PART I: Software Installation

1. **Install Python:**
    - Download Python 3.11.4
    - Recommended version: python-3.11.4
2. **Install Node.js:**
    - Download Node.js v18.16.1
    - Recommended version: node-v18.16.1
    - **Execute after installation:**
        - (CMD) Run: **npm install --global http-server**
        - (CMD) Type "http-server" to check if working.
            - If "Windows Security Alert" window pops, click "Allow access"
        - Execute below ONLY IF "http-server" is not working.
            - (BROWSER) Empty cache & hard reload page (dev tools, ctrl + right click)
3. **Install required libraries via CMD:**
    - 'pip install customtkinter'
    - 'pip install packaging'
    - 'pip install Pillow'
    - 'pip install google-auth'
    - 'pip install google-api-python-client'
    - 'pip install pywin32'
4. **Download GConnect App:**
    - Download GCONNECT
    - Click Code > Download Zip
    - Extract main folder to your Desktop.
    - Rename main folder to "GCONNECT"
5. **Launch the app** through "LaunchPad.bat" or the "GConnect App" shortcut.
    - If "Windows protected..." SmartScreen window pops, click "More info" > Run anyway.

PART II: Omada Cloud Controller Setup

1. **Omada Cloud Controller Account:**

   Please note this is not tested using SDN Controller and OC200 CSV file.

   - Login to your Omada Cloud Controller
   - Launch your Controller
   - Go to Settings (Global) > Export Data:
     - Export List: Voucher Codes
     - Format: CSV
     - Portal: "Your Voucher WiFi Name"
     - Save file to: “%USERPROFILE%\Desktop\GCONNECT\raw_csv”.

PART III: Google Account Setup

1. **Google Account:**

   2.1 **Access your Google Account:** Login to your Google account.

   - **Google Drive API Configuration:**

   2.2 **Enable Google Drive API:**

   - Visit Google Cloud Console.
   - Enable the Google Drive API.

   Project Creation and API Activation:

   2.3 **Create a New Project:**

   - Create a new project, using your voucher Wi-Fi portal name.

   2.4 **Enable Google Drive API:**

   - Go to APIs & Services.
   - Click +ENABLE APIS & SERVICES.
   - Select Google Drive.
   - Enable it.

   2.5 **Credentials Configuration:**

   2.5.1 **Creation of Service Account:**

   2.5.1.1 Go to Credentials (with the key icon).
   2.5.1.2 Click +CREATE CREDENTIALS.
   2.5.1.3 Select Service Account.

2.5.1.4 Provide Account Name (OPTIONAL).
2.5.1.5 Give Account ID (REQUIRED).
2.5.1.6 Add Description (OPTIONAL).
2.5.1.7 CREATE AND CONTINUE.
2.5.1.8 Assign it an "OWNER" role.
2.5.1.9 DONE.
2.5.1.9.1 Click on the newly created Service Account.
2.5.1.9.2 Go to the KEYS tab.
2.5.1.9.3 Click ADD KEY.
2.5.1.9.4 Create a new key.
2.5.1.9.5 Choose Key type as JSON.
2.5.1.9.6 CREATE (a file will be downloaded).
2.5.1.9.7 Rename the file to "service_account" (.json).
2.5.1.9.8 Save it to “%USERPROFILE%\Desktop\GCONNECT”.

2.5.2 **Creation of API Key:**
2.5.2.1 Go back to +CREATE CREDENTIALS.
2.5.2.2 Select API KEY.
2.5.2.3 Copy the API KEY.
2.5.2.4 Click "Edit API key."
2.5.2.5 Under API restrictions, select "Restrict API key."
2.5.2.6 Check Google Drive API > OK > SAVE.
2.5.2.7 Paste the API KEY into “%USERPROFILE%\Desktop\GCONNECT\put_api_key_here.txt”.

2.6 **GCONNECT App Configuration:**

2.6.1 Click the "RUN INITIAL CONFIG" button in the GConnect App.
2.6.2 Wait for the browser to open, displaying the macro file and voucher folder.

2.7 **Macro File Transfer to Android Phone:**

2.7.1 Download the macro file and transfer it to your Android phone.

Please note that the following steps are applicable only when using the same Google Account on both your PC and Android device.

2.7.2 Launch the Google Drive App on your Android device.

2.7.3 Go to the “Shared” tab > Swipe down to refresh screen.

2.7.3.1 GCONNECT folder should appear IF NOT procced to **STEP: 3.2.1.**

2.7.4 Navigate inside the “GCONNECT” folder.

2.7.5 Tap the more option (⋮) beside the macro file.

2.7.6 Select “Open with” (should open with MacroDroid automatically).

2.7.7 Proceed to **STEP: 3.2.6.**

PART IV: Android Device Setup

3. **Android Device (Voucher Server Phone):**

Before proceeding, ensure that your Android device meets the following requirements:

- The phone must not be rooted.
- It should have a SIM card, available service, and SMS credits (load for texting).
- Always connected to WiFi.
- Must have a separate GCash account installed.
- Should be able to receive GCash real-time notifications.

Configuration Steps for the Android Device:

3.1 **Install MacroDroid:**

- Install the MacroDroid app from the Google Play Store.
- Recommended permissions for MacroDroid:
    - Give app admin rights.
    - Disable battery optimization.
    - Allow background autostart.
    - Grant app notification access.
    - Lock app in recent app tray.

3.2 **Import Macro to MacroDroid:**

3.2.1 Launch the MacroDroid app.
3.2.2 Tap on Export/Import from the HOME screen.
3.2.3 Under Import, tap on Storage.
3.2.4 Browse and select the downloaded macro from your local storage.
3.2.5 Open the imported macro.
3.2.6 Tap on the Webhook trigger.
3.2.6.1 Tap Configure.
3.2.6.2 Copy the URL and then go back.
3.2.6.3 Send the URL to PC (looks like this => https://trigger.macrodroid.com/DEVICE_ID/sync_voucher).
3.2.6.4 (PC) Paste the URL into this text file => "%USERPROFILE%\Desktop\GCONNECT\put_md_url_here.txt".
3.2.7 Tap the import button (≡+) and enable the macro.
3.2.8 (GCONNECT APP) Click EXTRUP + SYNC.

PART V: Testing

4. **Perform Testing:**
    - Send a GCash amount (5.00) to the Server Phone's GCash number.
        - Sender should receive WiFi Voucher via SMS.

ADD-ON: Ideal Voucher Printing Preferences

These preferences are optimized to print multiple vouchers efficiently.

- **Layout:** Landscape (prints more vouchers than portrait)
- **Paper Size:** Letter (short)
- **Margin:** None
- **Scale:** Custom (93%)
- **Headers and Footers:** Unchecked (disabled)
- **Background Graphics:** Checked (only if you have a logo)

Note: This setup can print up to 32 vouchers on a single page, providing optimal efficiency for voucher printing.