



Guidebook : Data Integration and Automation with Python

ATASO

CBIC Planner

© 2025 CBIC Planner Division. All rights reserved.

This guidebook, including all content, design, and illustrations, is protected under copyright law.

Reproduction or distribution without prior written permission from the publisher is prohibited.

List Of Contents

Introduction 1

Workflow System Flowchart 2

Installing Tools and Libraries

Install VS Code 3

Install Python 4

Install Extensions 5

Install Python Libraries 7

Install File Repository 9

System Usage Guidelines

SSO Setup Guide 10

Power BI Setup Guide 18

Demo SSO 21

3rd Party Setup Guide 22

Demo 3rd Party 24

Automation New SSO Result

Setup Guide 25

Mini Data Analytics

Setup Production Report 27

Setup Output & Start Proses 28

Setup Power BI 30

Introduction

System Description

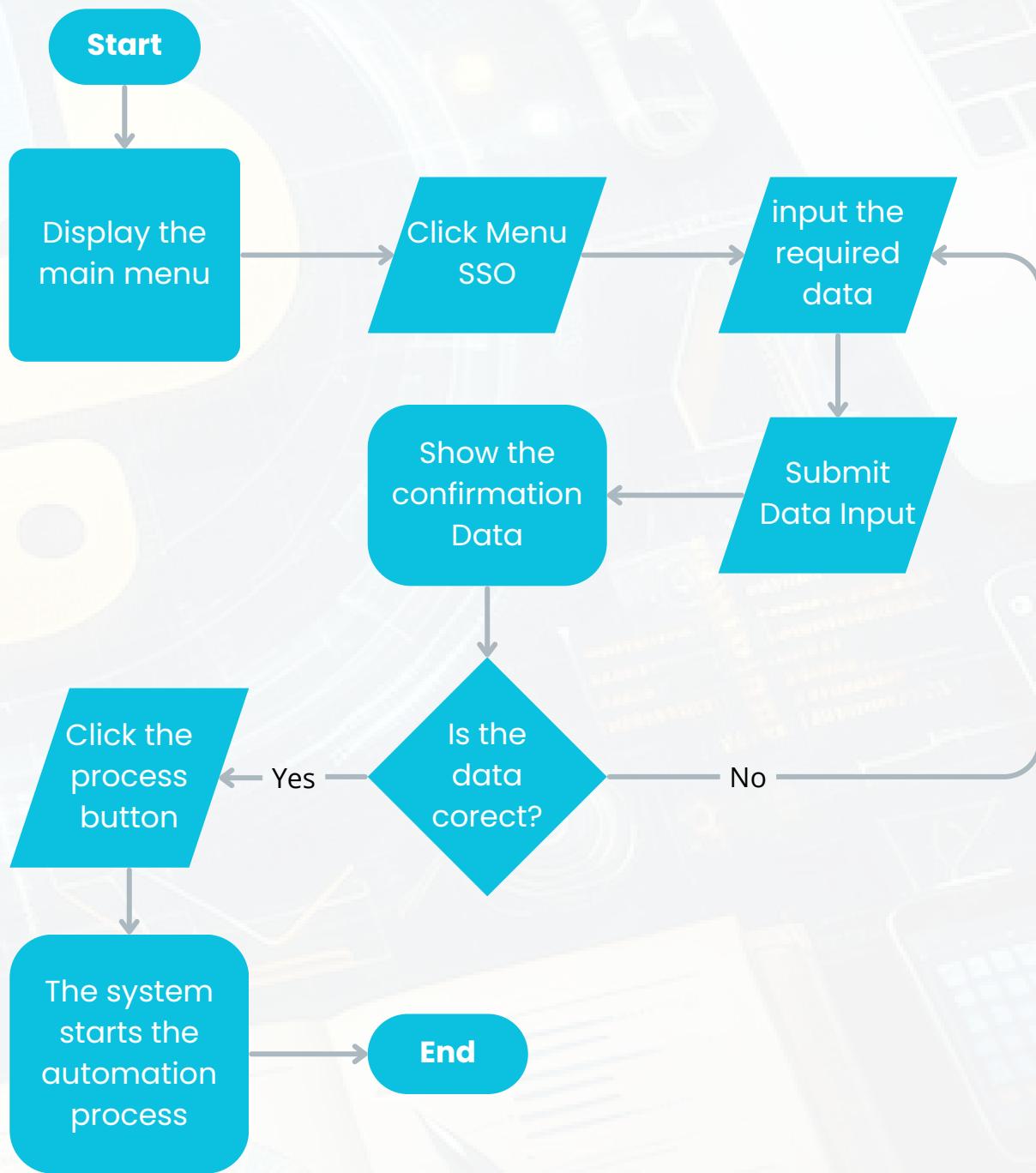
This automation system is a graphical user interface (GUI) based application developed using the Python programming language. The system is designed to facilitate various data processing activities that are often complex, repetitive, and require high accuracy. By combining automation functionality and ease of use through an intuitive GUI display, this application is intended to help users from various technical backgrounds to complete work more quickly and efficiently.

System Development Objectives

The main objective of developing this automation system is to improve productivity and quality in data processing. In more detail, the development of this system is aimed at:

- Save time and effort in completing repetitive work, so that users can focus on more strategic tasks.
- Provide a friendly and easy-to-understand user interface, allowing users of varying levels of technical expertise to operate the system without requiring in-depth training.
- Improves accuracy and consistency of results, as the entire automation process is systematically executed based on predefined logic and parameters.

Workflow System Flowchart



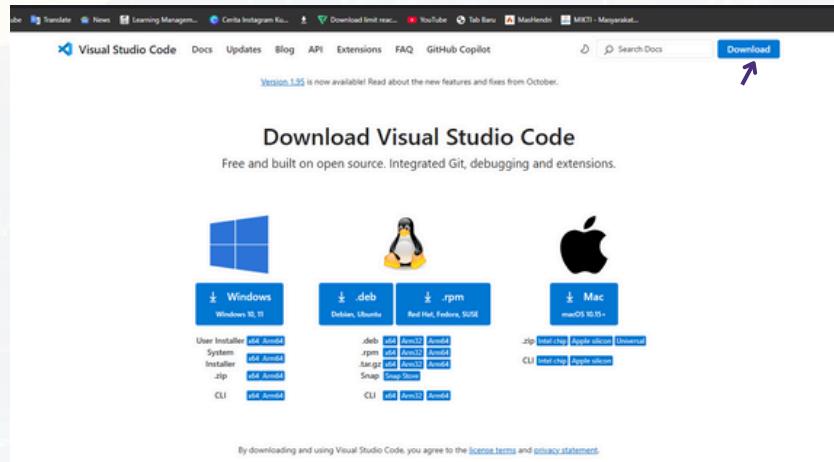
Installation and Setup

1. Installing Visual Studio Code (VSCode) & Extensions

Visual Studio Code (VSCode) is a lightweight yet feature-rich code editor, ideal for developing Python projects. Here is a step-by-step guide to installing VSCode.

- **Download Visual Studio Code**

- > Visit the official Visual Studio Code website at <https://code.visualstudio.com/>.
- > Click the Download for [Your OS] button, according to your operating system (Windows, macOS, or Linux).



- **Install VSCode**

- > **Open Installer:** Once the download is complete, open the installation file.
- > **Accept Agreement:** Select I accept the agreement and click Next.
- > **Choose Install Location:** Select the installation location (default: C:\Program Files\Microsoft VS Code), then click Next.
- > **Select Additional Tasks:** Check the following options for the best experience
 - Add "**Open with Code**" action to Windows Explorer file context menu
 - Add "**Open with Code**" action to Windows Explorer directory context menu.
 - Add to **PATH (required for command line)**.
 - Click **Next** to continue.
- > **Install:** Click Install to begin the installation process.

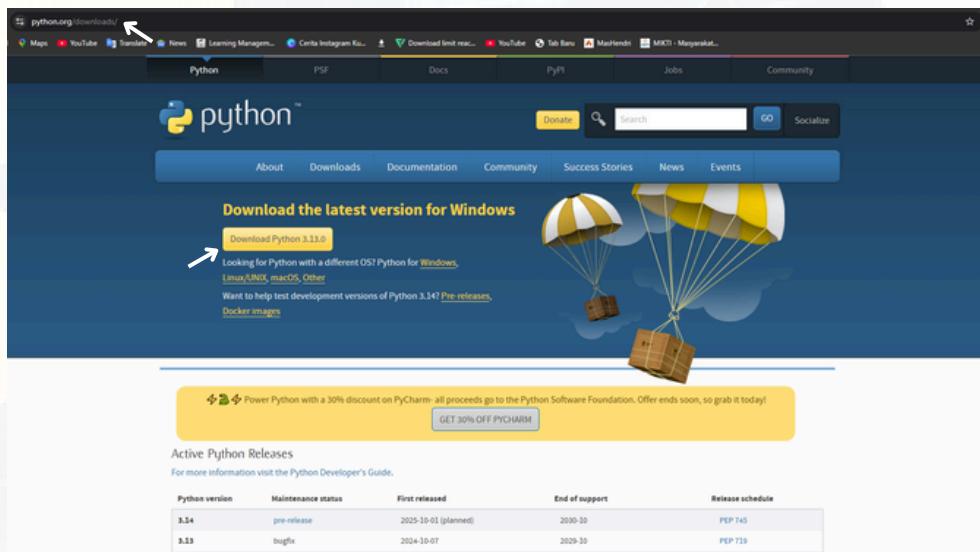
Installation and Setup

2. Installing Python

Python is the programming language that will be used in this guide. Here are the installation steps:

- **Download Python**

> Visit python.org/downloads and download the latest version



- **Install Python**

> Run the installer file, check the "**Add Python to PATH**" option, and then click Install Now.
> Verify Installation: Open a terminal or command prompt and type **python --version** to ensure Python is installed correctly.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.5011]
(c) Microsoft Corporation. All rights reserved.

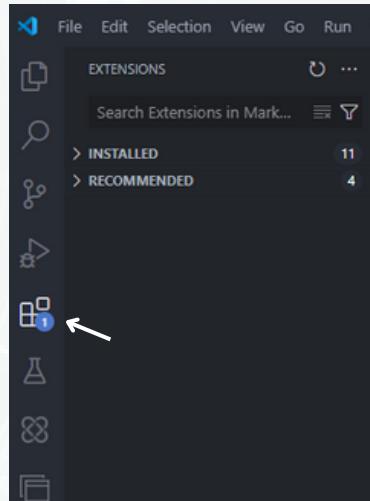
C:\Users\v_ian_barus>python --version
Python 3.13.0
```

Installation and Setup

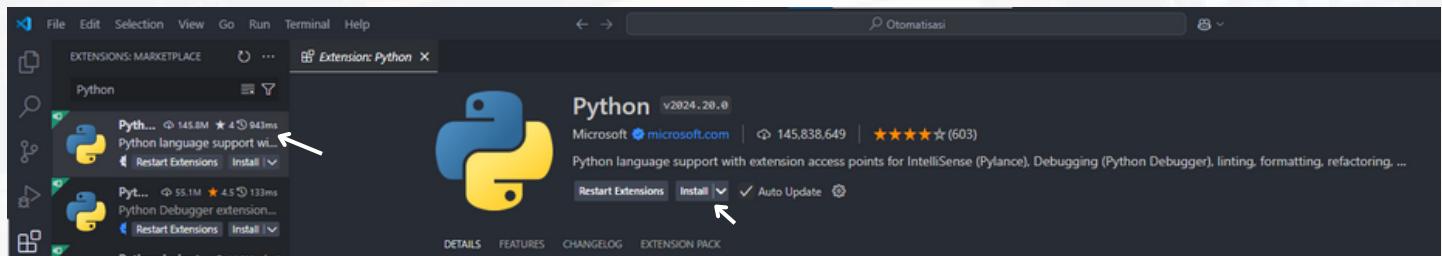
3. Installing Extensions

Steps to Add Extensions in Visual Studio Code:

- **Open VSCode**
- > **Launch** the Visual Studio Code app on your device and **click the Extensions icon** (the cube) on the left side.



- **Find the Python Extension:**
- > **Type Python** in the **search bar** at the top of the Extensions tab.
- > **Select** the extension made by **Microsoft** and click the **Install** button.
- > **Wait** for the installation process to **complete**.
- > This extension will add **debugging**, **linting**, and **Jupyter notebook** support to Python in VSCode.



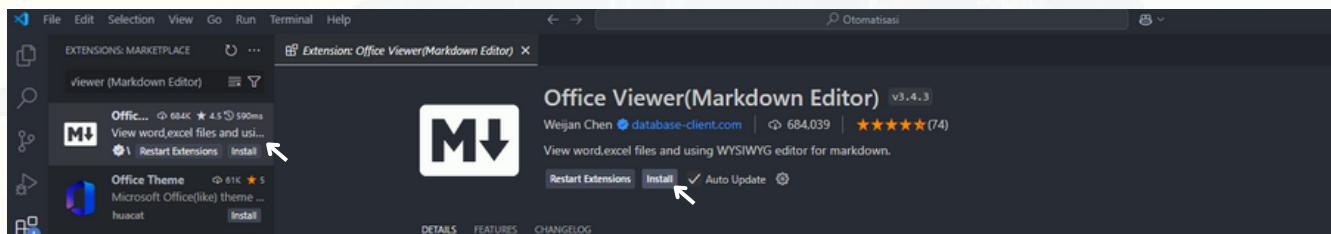
Installation and Setup

3. Installing Extensions

Steps to Add Extensions in Visual Studio Code:

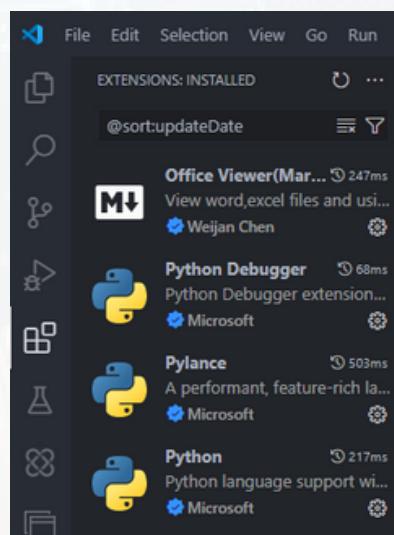
- **Find the Office Viewer (Markdown Editor) Extension:**

- > Type **Office Viewer** in the **search bar** at the top of the Extensions tab.
- > Select the extension **Office Viewer (Markdown Editor)** and click the **Install** button.
- > Wait for the installation process to **complete**.
- > This extension **makes it easy to edit Markdown files** with a **live preview feature**.



- **Find the Office Viewer (Markdown Editor) Extension:**

- > Once the installation is complete, **make sure the extension is active**. You can see it in the Installed Extensions section of the **Extensions tab**.
- > If there are any issues, click the settings icon on the extension and select **Reload Window** to reload VSCode.



- **Tips for Getting the Most Out of Extensions**

- > **Python:** Make sure you have **Python installed** on your device and set up the Python interpreter in VSCode for this extension to work optimally.
- > **Office Viewer:** Use this extension to create or edit Markdown documents with easy-to-access preview features.

Installation and Setup

4. Installing Library Python

Here are the steps to install the required Python libraries directly through VSCode using **pip**:

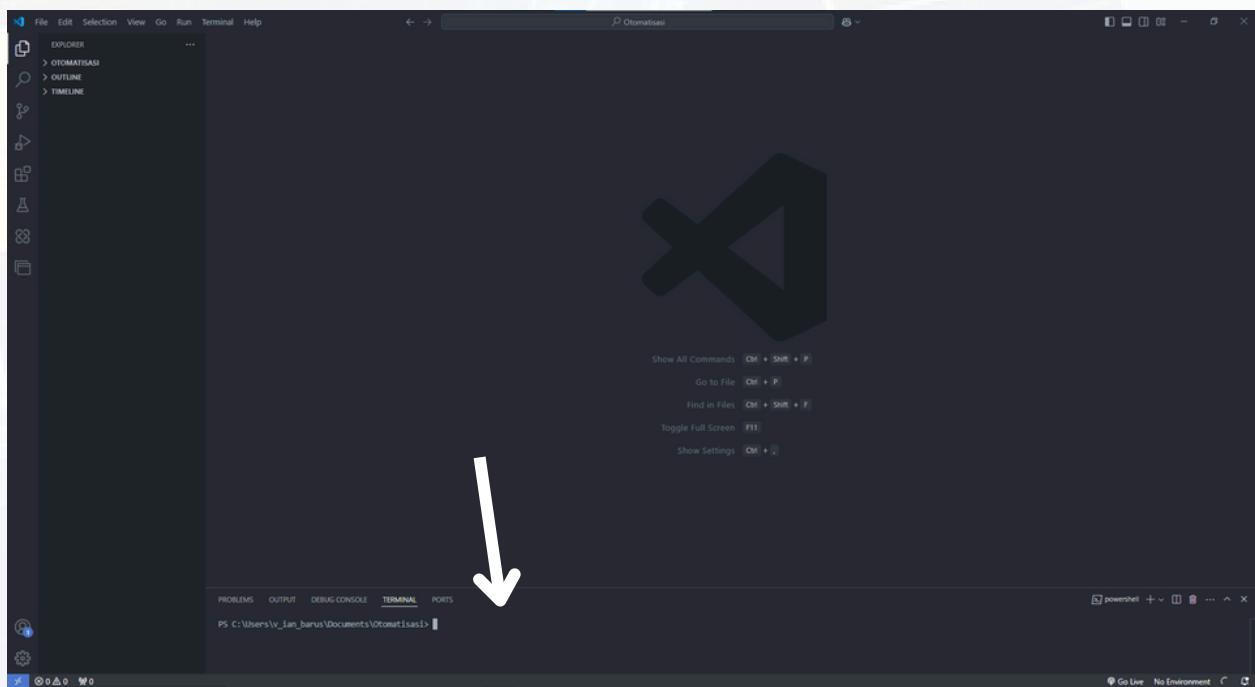
A. Open Terminal in VSCode, To open Terminal:

- Click the View menu at the top, then select Terminal.

Alternatively, use the keyboard shortcut:

- **Ctrl + ~** (Windows/Linux) or **Cmd + ~** (Mac).

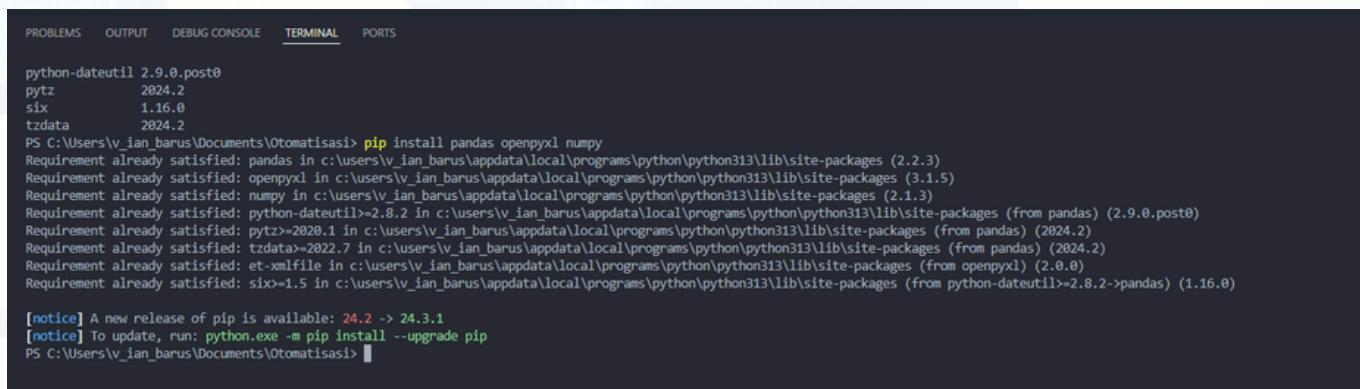
The Terminal will open **at the bottom** of the VSCode screen.



Installation and Setup

B. Install Libraries with pip:

- Type the following command in the terminal to **install the library**:
 - **pip install pandas openpyxl numpy**

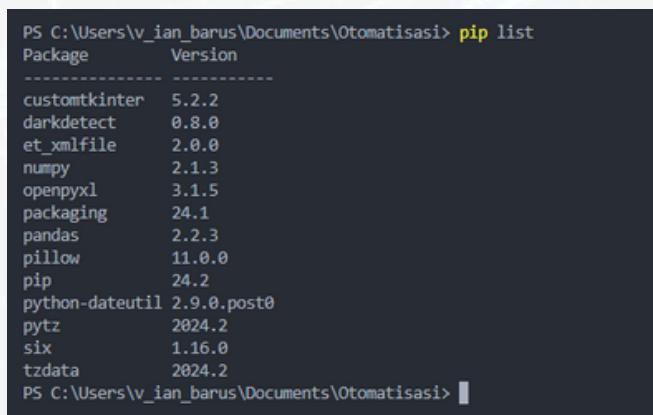


```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

python-dateutil 2.9.0.post0
pytz 2024.2
six 1.16.0
tzdata 2024.2
PS C:\Users\v_ian_barus\Documents\Otomatisasi> pip install pandas openpyxl numpy
Requirement already satisfied: pandas in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (2.2.3)
Requirement already satisfied: openpyxl in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (3.1.5)
Requirement already satisfied: numpy in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (2.1.3)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: et-xmlfile in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (from openpyxl) (2.0.0)
Requirement already satisfied: six>=1.5 in c:\users\v_ian_barus\appdata\local\programs\python\python313\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)

[notice] A new release of pip is available: 24.2 -> 24.3.1
[notice] To update, run: python.exe -m pip install --upgrade pip
PS C:\Users\v_ian_barus\Documents\Otomatisasi>
```

- Once the installation is complete, **make sure** the library is **installed** by **running** the following command:
 - **pip list**



```
PS C:\Users\v_ian_barus\Documents\Otomatisasi> pip list
Package           Version
-----
customtkinter    5.2.2
darkdetect       0.8.0
et_xmlfile       2.0.0
numpy            2.1.3
openpyxl         3.1.5
packaging        24.1
pandas           2.2.3
pillow           11.0.0
pip              24.2
python-dateutil  2.9.0.post0
pytz             2024.2
six              1.16.0
tzdata           2024.2
PS C:\Users\v_ian_barus\Documents\Otomatisasi>
```

*Brief Explanation of Library Use

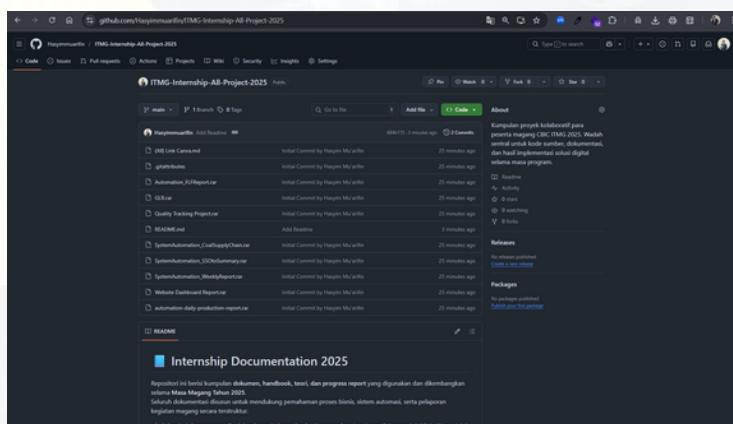
- **Pandas:** Processing and analyzing table data such as Excel/CSV.
- **Openpyxl:** Reading, writing, and modifying Excel files (.xlsx).
- **NumPy:** Mathematical operations and array manipulation for numeric data.

Installation and Setup

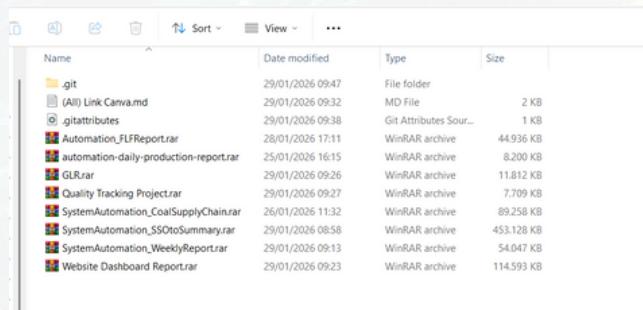
5. Install File Repository From Github:

download the system files by going to the following link on GitHub:

<https://github.com/Hasyimmuarifin/ITMG-Internship-All-Project-2025>



Extract the downloaded ZIP file. Once extracted, you will see several files inside as mentioned below



Next, open the Terminal or Command Prompt (CMD), then point the working directory to the previously extracted system folder. then install all the dependencies required by the system by running the following command in Terminal or CMD: pip install -r requirements.txt.

```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.26100.3915]
(c) Microsoft Corporation. All rights reserved.

E:\FILE KERJA ITM\Project\Coba\ITM-CBIC_Excel-Automation-System-main>pip install -r requirements.txt
```