

**Intelication**

**Technical Document**



Contents

[Development Tools 3](#_Toc190280187)

[Development assembly: 3](#_Toc190280188)

[Architecture 3](#_Toc190280189)

[Application Architecture Overview 3](#_Toc190280190)

[Application Workflow 4](#_Toc190280190)

[Project Setup & Execution Steps 5](#_Toc190280190)

# Development Tools

## Development assembly:

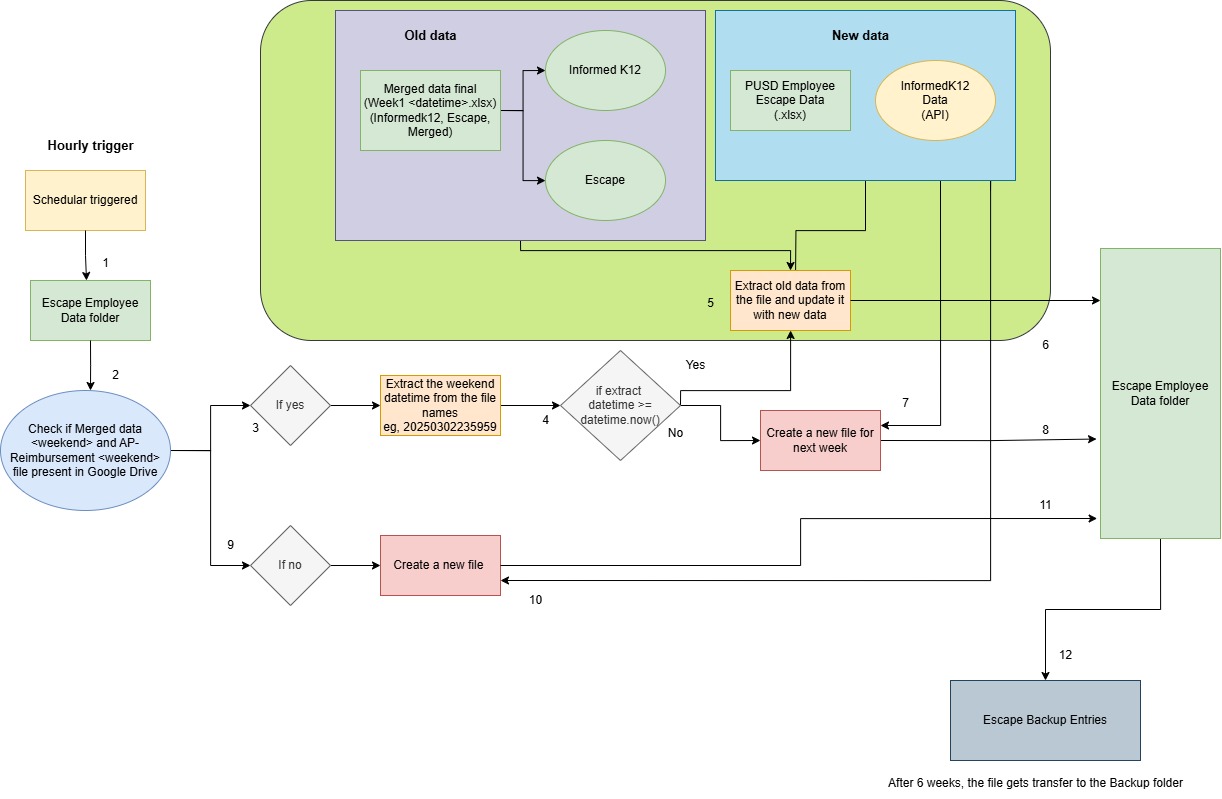
VS Code

## Programming Language:

Python

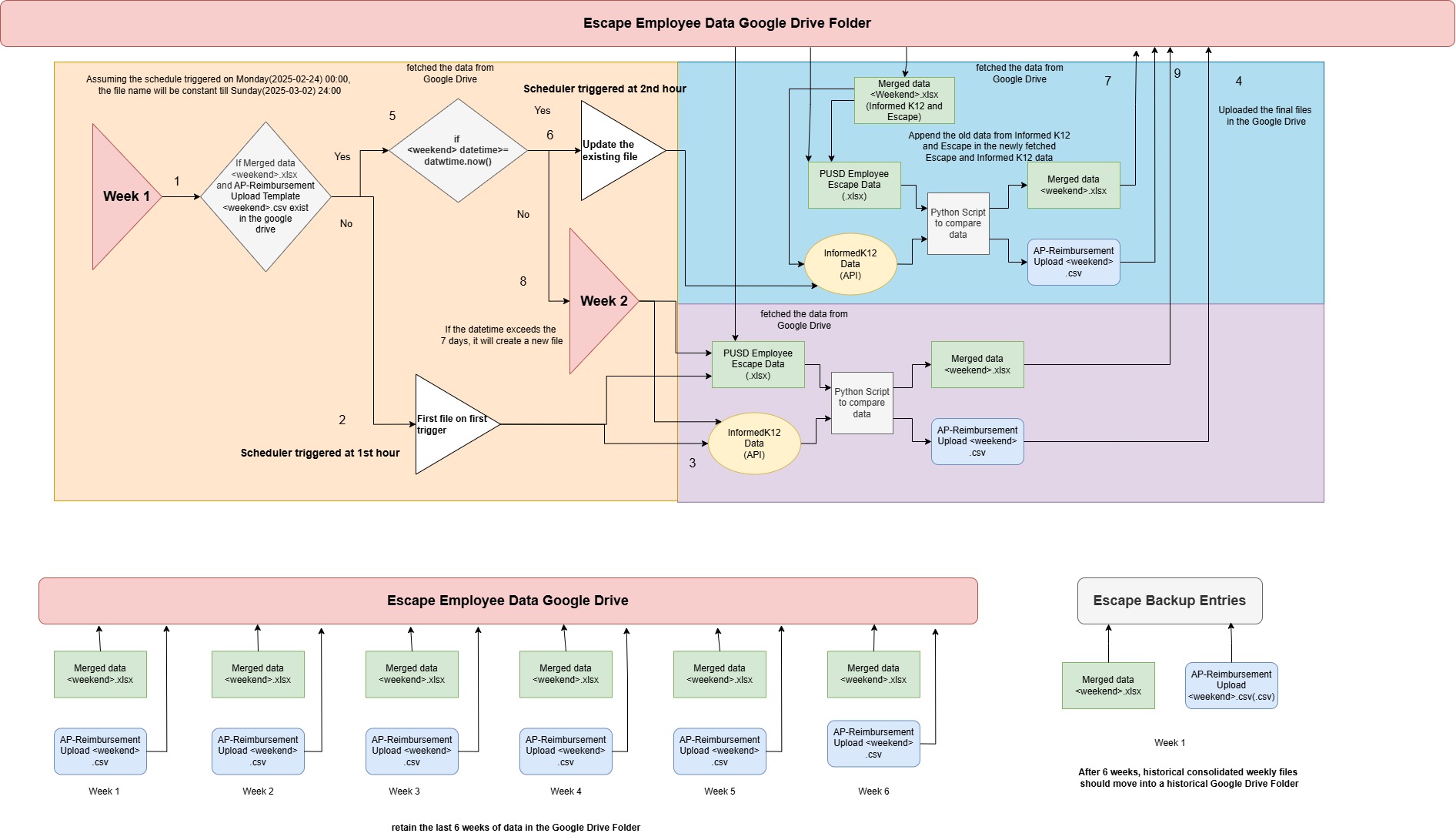
# Architecture

## Application Architecture Overview



1. The scheduler triggers and executes the script every hour, checking the "Escape Employee Data" folder in Google Drive for any existing files. To connect Google Drive with the Python script, the Google Drive API must be enabled, a service account must be created in Google Cloud, and a JSON key must be generated for authentication.
2. In the "Escape Employee Data" folder, the script checks if the base file names "Merged Data" and "AP-Reimbursement Upload" exist.
3. If they exist, the script fetches the files Merged Data <weekend>.xlsx and AP-Reimbursement Upload <weekend>.csv and extracts the <weekend> datetime from their names.
4. It then checks whether the <weekend> datetime is greater than or equal to today’s datetime.
5. If yes, the script extracts old data (Informed K12 and Escape data) from the uploaded Merged Data <weekend>.xlsx file in Google Drive, fetches new data from the Informed K12 API and PUSD Employee Escape Data, updates the old data with the new, and performs a comparison.
6. The updated file is then uploaded back to the "Escape Employee Data" Google Drive folder.
7. If the <weekend> datetime is not greater than or equal to today’s datetime, the script fetches new data from the Informed K12 API and PUSD Employee Escape Data and performs a comparison.
8. The updated file is then uploaded to the "Escape Employee Data" Google Drive folder.
9. If the "Escape Employee Data" Google Drive folder does not contain the base file names "Merged Data" and "AP-Reimbursement Upload," the script proceeds with fetching new data.
10. It retrieves the new data from the Informed K12 API and PUSD Employee Escape Data and performs a comparison.
11. The updated file is then uploaded to the "Escape Employee Data" Google Drive folder.
12. Once the file is uploaded, it will be automatically transferred to the "Escape Backup Entries" Google Drive folder after six weeks

## Application Workflow



1. On the first trigger, the script checks if the base file names "Merged Data" and "AP-Reimbursement Upload" exist in the "Escape Employee Data" folder in Google Drive.
2. If they do not exist, a new file is created and uploaded to the "Escape Employee Data" folder.
3. The script fetches PUSD Employee Escape data from Google Drive and Informed K12 data from the API. It then compares the Informed K12 data with Escape data using the Email column as a unique identifier.

* If the email matches, other columns are checked.
* If all columns match, no highlight is applied.
* If there is a mismatch, incorrect data is replaced with Escape data and highlighted in yellow.
* If an email is not found in Escape data, it is highlighted in red.
* At the end, a Merged Data <weekend>.xlsx file is created containing three sheets: Informed K12 Data, Escape Data, and Merged Data. Additionally, an AP-Reimbursement Upload <weekend>.csv file is generated.

1. The Merged Data <weekend>.xlsx and AP-Reimbursement Upload <weekend>.csv files are uploaded to the "Escape Employee Data" folder in Google Drive.
2. If the files already exist, the script fetches Merged Data <weekend>.xlsx and AP-Reimbursement Upload <weekend>.csv, extracts the <weekend> datetime, and checks if it is greater than or equal to today’s date.
3. If the <weekend> datetime is greater than or equal to today’s date, the script fetches old Informed K12 and Escape data from the existing Merged Data <weekend>.xlsx file. It then retrieves new PUSD Employee Escape data from Google Drive and Informed K12 data from the API. The new data is appended to the old data, and merging and comparison are performed.
4. The updated Merged Data <weekend>.xlsx and AP-Reimbursement Upload <weekend>.csv files are uploaded to the "Escape Employee Data" folder in Google Drive.
5. If the <weekend> datetime is not greater than or equal to today’s date, it signifies the start of a new week. A new file is created using PUSD Employee Escape data and Informed K12 data from the API, and merging and comparison are performed.
6. Both new files are uploaded to the "Escape Employee Data" folder in Google Drive.
7. By the end of the sixth week, the "Escape Employee Data" folder will contain 12 files (Merged Data <weekend>.xlsx and AP-Reimbursement Upload <weekend>.csv for each week).
8. After the sixth week, the first week’s files will be transferred to the "Escape Backup Entries" folder in Google Drive.

**Project Setup & Execution Steps:**

Python version: 3.13.2 (Latest stable version)

Commands:

1. **Create virtual environment :**     python -m venv venv
2. **Activate the virtual environment:**     venv\Scripts\activate.ps1
3. **Install libraries from requirements file:**  
        pip install -r requirements.txt
4. **Final command to run the application:**  
        python main.py





21, IT Park, Gayatri Nagar, Behind VNIT, Nagpur – 440022

**Contact Us:**

Mob.: India - +91 8007861579, Dubai - +971 508794909

Web: [www.nicesoftwaresolutions.com](http://www.nicesoftwaresolutions.com)

E-Mail: info@nicesoftwaresolutions.com

**Thank**

**You**