

# Technical Design Document: Google Takeout Viewer Suite

---

## 1. Project Overview

**Google Takeout Viewer Suite** is a collection of three independent, standalone Python applications designed to parse and visualize data exported from Google Takeout. Each application functions as a distinct tool with its own entry point and full GUI, replicating the native experience of smartphone or desktop productivity apps.

## 2. Technology Stack

- **Language:** Python 3.10+
- **IDE:** PyCharm
- **GUI Framework: PyQt6** (Chosen for robust widget support, HTML rendering for emails, and modern styling capabilities).
- **Data Parsing:**
  - `vobject` : For robust VCF (Contact) and ICS (Calendar) parsing.
  - `mailbox` : Standard Python library for MBOX handling.
  - `email` : Standard Python library for MIME processing.
  - `BeautifulSoup4` (optional but recommended): For sanitizing email HTML.

## 3. Modular File Structure (Standalone Architecture)

The structure is designed so that each viewer folder is self-contained. You can run `main.py` inside each folder to launch that specific tool immediately.

```
GoogleTakeoutSuite/
|
|— common/                # Shared resources across apps
|   |— __init__.py
|   |— styles.qss          # Unified theme (dark/light mode, CSS styli
|   |— utils.py            # Shared helper functions (date formatting,
|
|— app_contacts/          # Standalone Contact Viewer
```

```
| |— __init__.py
| |— main.py                # Entry point for Contact Viewer
| |— ui_layout.py          # GUI setup (Splitter, Lists, Labels)
| |— parser.py             # VCF parsing logic
|
|— app_calendar/           # Standalone Calendar Viewer
|   |— __init__.py
|   |— main.py             # Entry point for Calendar Viewer
|   |— ui_layout.py        # GUI setup (Calendar Widget, Event List)
|   |— parser.py           # ICS parsing logic
|
|— app_mail/               # Standalone Mail Viewer
|   |— __init__.py
|   |— main.py             # Entry point for Mail Viewer
|   |— ui_layout.py        # GUI setup (3-pane layout, HTML Viewer)
|   |— parser.py           # MBOX parsing & email body extraction
|
|— launcher.py             # (Optional) A unified menu that launches t
```