

Mac Workflow for Tracking Daily Research Progress

Ronald (Ryy) Glenn Thomas

2025-02-09

Table of contents

1	Introduction	2
2	Mac Workflow for Tracking Daily Research Progress	2
2.1	1. Folder Structure	2
2.2	2. Daily Logging System	2
2.2.1	Automated Entry Creation	2
2.3	3. Dictation & Summarization via Chat-GPT	3
2.3.1	Dictate Notes to ChatGPT	3
2.3.2	ChatGPT Summarization	4
2.4	4. Version Control (Git)	4
2.4.1	Automate Git Commit	4
2.5	5. Searching & Reviewing	4
2.5.1	Find past notes by project:	4
2.5.2	Convert Logs to PDF for Reports	5
2.6	6. Syncing Across Devices	5
2.7	Summary of the Workflow	5



1 Introduction

2 Mac Workflow for Tracking Daily Research Progress

This workflow **organizes, records, and automates** progress tracking while integrating ChatGPT for **dictation and summarization**.

2.1 1. Folder Structure

Create a structured directory for all research projects:

```
mkdir -p ~/prj/ResearchLog ~/prj/{X1,X2,X3,X4,X5,X6,X7,X8,X9,X10}
```

Each project folder contains:

- `notes.md` → Running log of project progress
- `references.bib` → Citation management
- `data/` → Datasets and related files
- `scripts/` → Analysis and code
- `project_title_with_investigator_name/` → Empty file to facilitate fzf search

2.2 2. Daily Logging System

- **Centralized daily log:** `~/prj/daily_log.md`
- Uses **ChatGPT summaries** for dictated notes.

2.2.1 Automated Entry Creation

To append a new daily entry:

```
echo "### $(date +%Y-%m-%d)" >> ~/Documents/ResearchLog/daily_log.md
echo "- X1: " >> ~/Documents/ResearchLog/daily_log.md
echo "- X2: " >> ~/Documents/ResearchLog/daily_log.md
```

Alternative: Use a Vim template:

```
autocmd BufNewFile *ResearchLog/daily_log.md Or ~/Templates/daily_template.md
```

(Pre-fills the structure upon opening the log file.)

2.3 3. Dictation & Summarization via ChatGPT

2.3.1 Dictate Notes to ChatGPT

- **Use Mac's Dictation Feature:**
 - Enable by pressing **fn** (function key) **twice**.
 - Dictate notes for each project.
 - Example:

“For X3, I explored theorem 2 in Joe’s 1987 paper and identified a simplified approach for $(R \times 2)$ tables.”
 - Suggested prompt: “I’m about to dictate research experiment notes. When I’m done, provide a medium-level verbose summary. The notes start here: [dictate notes] ... The notes end here.”
 - run the command:

```
./research_update
```

This script contains the following code along the date and project name:

```
echo "$ (pbpaste)" >> ~/prj/daily_log.md
```

2.3.2 ChatGPT Summarization

- Summarized daily research logs (like the example you provided) will be **formatted and ready for inclusion** in `daily_log.md`.
-

2.4 4. Version Control (Git)

To **track changes** and **ensure backups**:

```
cd ~/Documents/ResearchLog
git init
git add daily_log.md
git commit -m "Daily log update $(date +%Y-%m-%d)"
git push origin main
```

2.4.1 Automate Git Commit

Add this to crontab for daily commits:

```
0 23 * * * cd ~/prj/research_update && git add . && git commit -m "Daily log $(date)" && git push
```

(Runs at 11 PM daily.)

2.5 5. Searching & Reviewing

2.5.1 Find past notes by project:

```
grep "X3" ~/prj/research_update/daily_log.md
```

2.5.2 Convert Logs to PDF for Reports

```
pandoc ~/prj/research_update/daily_log.md -o ~/prj/research_update/daily_log.pdf
```

(Creates a readable summary document.)

2.6 6. Syncing Across Devices

- Use **iCloud Drive**, **Dropbox**, or **GitHub** for remote access.
- Optionally, use **rsync** for local backups:

```
rsync -av ~/Documents/ResearchLog ~/iCloudDrive/ResearchBackup/
```

2.7 Summary of the Workflow

1. **Dictate notes** via Mac's built-in dictation feature.
2. **ChatGPT summarizes** and formats them.
3. **Paste summary into daily_log.md.**
4. **Git commits and syncs logs** every day.
5. **Easily search, review, and convert logs to PDFs.**

Would this workflow fit your needs? Let me know if you need refinements!