Sprint 3

📄 Sprint 3 – Implementation Log – *Personal Task Manager*

🎯 Sprint Goal

> Improve task navigation, clarity, and responsiveness by implementing search, workload indicators, and real-time updates across the task interface.

📝 User Stories Delivered

- As a user, I want to filter tasks by priority or search by keyword so I can find important tasks faster.
- As a user, I want to see how many tasks I have left to complete so I can better manage my time.
- As a user, I want tasks to appear instantly after adding so I don't have to refresh manually.

⚙️ Technical Implementation

🔍 PTM7 – Search by Task Title

- Search input added to top of task list
- `searchTerm` bound with `[(ngModel)]`
- Dynamic filtering in `getTasksByStatus()` based on case-insensitive match

📝 PTM8 – Remaining Tasks Counter

- `getRemainingTasksCount()` added to count tasks not marked as "Done"
- Count displayed dynamically in UI, updated on status change

⚡ PTM9 – Instant Task Refresh (No Reload)

- Angular `signal` (`taskRefreshSignal`) introduced for real-time task syncing
- Signal triggered in `task-form.component.ts` after adding a task
- `task-list.component.ts` uses `effect()` in constructor to listen and reload list

📎 Related Links

- 🔗 GitHub Repo
- 🔗 Sprint 3 Planning
- 🔗 Sprint 3 Retrospective
- 🧠 Signal file: `src/app/signals/task-refresh.signal.ts`

🧠 Notes

- Search and task counter are lightweight but enhance usability
- Signal-based architecture kept component decoupling clean
- Priority filter postponed to Sprint 4 to keep sprint focused and manageable
- Feature additions were non-breaking and stable across all task states