

Main Flow Tightening and Timer Duration Picker Work Report

#Overview

Today's work focused on the next concrete step in the vision-gap plan after the already completed sections.

The main goal was to reduce friction in the path from choosing work to actually starting a focus session. That meant tightening the task-level and dashboard-level sprint actions, introducing a consistent default focus duration, and making the timer screen feel faster to enter without removing the older duration-picker path entirely.

Later in the same work session, the scope narrowed further into the timer screen itself because the reintroduced picker flow behaved incorrectly. That led to a smaller follow-up fix focused specifically on stabilizing the picker state and preventing the screen from resetting while the user scrolls.

The scope also expanded into the help-flow modal on the dashboard and subjects screens so its explanation of the app structure matches the way the app now actually works.

After that, one more timer-flow bug surfaced in the post-session overlay itself. A completed break could still reuse the focus-session action menu, which incorrectly offered the user another break instead of only the actions that make sense after a break has ended.

#ImplementedFeatures

#DefaultFocusDuration

Introduced a shared default session-duration source for the low-friction focus flow:

- added 'lib/sessionDefaults.ts'
- defined:
 - 'DEFAULT_FOCUS_DURATION_MINUTES'
 - 'DEFAULT_SHORT_BREAK_DURATION_MINUTES'
- reused those constants across the timer, task details, and dashboard paths

This removed the need to hardcode the same default duration in multiple places and made the main sprint path more consistent.

#TaskDetailsPrimarySprintAction

Updated the task details screen so 'Start Sprint' is the strongest action on the page:

- moved 'Start Sprint' out of the lower row of equal-weight controls
- made it the primary full-width action above 'Edit' and 'Delete'
- added small helper text clarifying that the action starts a '25' minute focus sprint
- updated the task-details start flow so it passes the default focus duration into the timer route
- tightened the active-session replacement alert text so it clearly states what will happen before the current session is replaced

This makes the task screen push the user more directly toward real study work instead of presenting sprint start as only one option among several management actions.

#DashboardDirectSprintStart

Reduced dashboard-to-timer friction for upcoming tasks:

- added a 'Start Sprint' action directly on the 'Tasks with upcoming deadlines' cards
- made that action open the timer immediately with the shared default focus duration
- handled the three relevant states:
 - no active session
 - an expired stored session
 - an already running different session that must be explicitly replaced
- renamed the active-session dashboard button from 'Open Session' to:
 - 'Resume Sprint'
 - 'Resume Break'

This removed one unnecessary detour where the user had to open task details first before reaching the timer.

#TimerDefaultDurationFlow

Changed the timer entry flow so focus sessions no longer force the user through duration selection before they can begin:

- changed the default focus-session setup to show a fixed default duration first
- kept break sessions on a fixed-duration path as before
- made the start action use the default focus duration immediately unless the user actively chooses a custom one

This better matches the low-friction part of the vision plan, where starting work should feel immediate rather than configuration-heavy.

#CustomDurationReturnPath

Reintroduced the old duration-picker flow as an explicit optional side path instead of the default:

- added a 'Choose a different duration' button on the pre-start focus timer screen
- reopened the old picker presentation only when the route enters an explicit picker mode

This keeps the faster default path while still preserving the older manual-duration interaction for users who want it, without adding a second reversal action inside the picker itself.

#PostSessionActionClarity

Adjusted the timer completion overlay so the focus-session exit path is more explicit:

- after a completed focus session, the overlay now offers:
 - 'Start short break'
 - 'Continue same task'
 - 'Back to dashboard'
- updated the explanation text so the available next actions are described directly in the overlay copy

This makes the post-session decision path closer to the plan's requirement that break, continue, and dashboard-return actions should be simple and explicit.

#TimerPickerGlitchFix

Fixed the first version of the restored duration picker after it showed unstable behavior:

- the picker numbers could initially appear blank until the list was scrolled
- the selected duration could snap back incorrectly when scrolling ended
- the cause was that the picker route was being rewritten while the user interacted with the list
- changed picker selection to use local component state instead of route replacement on every scroll stop
- added explicit initial offset restoration on picker open so the visible selection matches the current duration immediately
- kept the route change only for entering or leaving picker mode, not for every intermediate selection

This made the picker usable again without undoing the lower-friction default entry flow.

#HelpFlowAlignment

Updated the help modal so it matches the current app structure more closely:

- kept the main hierarchy as:
 - 'Subject'
 - 'Assignment'
 - 'Task'
 - 'Sprint'
- updated the 'Sprint' explanation so it now reflects the real post-session flow:
 - take a break
 - continue the same task
 - return to the dashboard
- changed the supporting copy so it explains that the work path now leads into both sprints and breaks instead of only into one focused work session
- added quick-map text clarifying the dashboard's current role:
 - resume active session
 - start next sprint

- review recent progress
- changed the help CTA on the dashboard from 'Start with Subjects' to 'Open Subjects'
- changed the help CTA on the subjects screen from 'Start with Subjects' to 'Close Guide'

This keeps the help flow aligned with the app's actual current behavior instead of leaving it stuck on an older sprint-only interpretation.

#PostBreakMenuFix

Fixed a timer completion bug where a finished break could still produce the same action menu as a finished focus session:

- the post-session overlay had to know which session type actually just ended
- the previous flow could fall back to local screen state instead of the persisted active session
- this caused a break completion to sometimes be treated like a focus completion
- changed the completion flow so it reads the stored active session before building the post-session prompt
- reused that same session snapshot when finalizing the session in Supabase

This means the overlay now behaves correctly after a break finishes:

- it does not offer 'Start short break' again
- it instead keeps the narrower break-finished path:
 - continue with the same task
 - go back to the dashboard

#ProblemsAndSetbacks

#PickerStateReset

The main issue during this work happened after the older picker screen was reintroduced as an optional path.

The first implementation reopened the picker route correctly, but it also updated the route params again when scrolling stopped. In practice this caused two visible problems:

- the initial number presentation was unstable
- the selected value could reset unexpectedly after momentum ended

The fix was to keep picker selection local to 'app/task/timer.tsx' while the picker is open, and only use route params to decide whether the picker mode should be shown in the first place.

#PostSessionTypeMismatch

Another issue appeared after the post-session focus actions were introduced.

The completion overlay already had separate UI for 'focus' and 'break' sessions, but the value used to choose between them was not robust enough. In practice, that made it possible for a finished break to reopen the focus-style menu and incorrectly offer another break.

The fix was to derive the completed session type from the persisted active session that had actually been running, rather than relying only on the screen's local state at the moment the animation finished.

#CurrentState

The timer/task/dashboard flow now does more to push the user into focused work with fewer unnecessary steps.

The app now supports:

- a shared default focus duration for the main sprint path
- a stronger 'Start Sprint' action on the task details screen
- direct sprint start from dashboard upcoming-task cards
- clearer 'Resume Sprint' and 'Resume Break' wording on the dashboard
- a fixed default-duration entry state on the timer screen
- an optional custom-duration picker path instead of a forced picker
- explicit post-focus next actions for break, continue, or dashboard return
- a stable picker implementation that keeps its selected value while the user scrolls
- a corrected break-finished overlay that no longer offers another pause when the completed session was already a break
- a help-flow explanation that now matches the real sprint, break, dashboard, and subjects flow more closely

At this point, the timer flow is more aligned with the vision requirement that starting work should feel fast, focused, and low-friction rather than like a chain of setup steps.

#Verification

Static checks were run after the implementation work and after the picker bug fix:

An additional static check was also run after the post-break menu fix:

```
““text  
npx tsc --noEmit  
exited successfully
```

```
npm run lint  
exited successfully
```

```
npx tsc --noEmit  
exited successfully  
““
```