Team roles and Responsibilities

Name	Role	Description
Daniel	API	CanvasClient; fetchCourses/Assignments(range)/Announcements(range); error mapping.
Amanda	Models/Data	CSV repos (upsert/query); Sorting/WeekUtil; `.profile` read/write + validation.
Quentin	UI	MainView + tabs/toolbar; tables w/ difficulty stars; banners & styles.
Kenny	Controller	Launch decision; Main state & providers from CSV; sorting + difficulty persist.

Cougar Planner Project Outline

Product Backlog

ID	Description	Dependencies
~~T01~~	~~Build .profile **reader** that loads exact keys, trims whitespace, and defaults missing keys to the spec's defaults. Provide getters for booleans as strings "true"/"false" and convenience accessors for parsed values. Surface a typed result with validation errors collected (not thrown) so UI can show them.~~	~~—~~
~~T02~~	~~Build .profile **writer** that persists keys in a fixed order with authToken on the first line. Writes atomically (temp →rename) and preserves string booleans. Provide an "overwrite fields" API to update one or many keys safely.~~	~~T01~~

~~T03~~	~~Implement CSV layer for courses.csv, assignments.csv, announcements.csv: read all, append, and **upsert by ID**. Normalize headers to snake_case; handle empty fields; ensure no duplicate rows. Expose simple repository-style APIs per file.~~	~~—~~
~~T04~~	~~Implement **date/time utilities**: ISO parsing for YYYY-MM-DD and HH:MM (24h), plus helpers for “start of week” per weekStart and computing current/selected week bounds. Include comparison helpers used by sorting.~~	~~—~~
~~T05~~	~~Validate `Profile` values (e.g., `weekStart`), replace invalid values with defaults, and add warning messages for UI display..~~	~~T01, T04~~
T06	Develop the first screen for Canvas token input, displaying a plain text storage warning. Show an error and prevent continuation if "store token = yes" is selected but the token field is empty. Do not print the token elsewhere.	T01, T05
T07	Create screens for all settings (week start, open to, default view, sort mode + order, store token, store data, show countdown) Save to `profile`.	T02, T06
~~T08~~	~~**Launch controller** implementing the decision tree (Orientation vs Token-Missing vs Enter Token vs Main). Reads .profile, routes accordingly, and passes computed week/day context.~~	~~T01, T04, T07~~
T09	~~Build Token-Missing and Enter Token flows: allow entering a token and saving to profile (respecting storeToken), or “Continue Offline” if any CSVs exist; on successful save, resume the normal launch path and initialize the Main view with computed period context.~~	T02, T08
~~T10~~	~~**Canvas API client**: shared HTTP client with Authorization header, base URL, timeout, and error mapping (401/403/429/5xx/timeout). No retries here; retries are policy-level.~~	~~—~~
~~T11~~	~~**Canvas Service**: fetch by **week range** (current or selected week) for Courses, Assignments, Announcements. Map API JSON → DTOs → repository models; return lists ready for merge.~~	~~T10, T04~~
~~T12~~	~~Implement a merge function to update CSV files by ID using fetched lists: update existing items, add new ones. For assignments, preserve local difficulty values. Produce a "what was new" list for UI.~~	~~T03~~

T13	Build the main screen shell with Announcements and Assignments tabs, a Week/Day toggle, and Prev/Today/Next buttons. Remember the selected week/day. Load and fetch data for a week/day the first time it's visited.	T08, T04
~~T14~~	~~Create "data provider" helper functions for each tab. For the current week/day, read CSVs, return relevant rows, look up `course_name` by `course_id`, and provide a friendly empty message if no data.~~	T03, T04, T13
T15	Add sorting "by date then time" for assignments: 1. `due_date` (earlier first, blank last), 2. `due_time` (earlier first, blank last), 3. `course_name` (A→Z), 4. `assignment_name` (A→Z). Include a small test list.	T04, T14
T16	Add sorting "by difficulty, then date/time" for assignments: 1. `difficulty` (ascending or descending, empty last), 2. `due_date` (earlier first, blank last), 3. `due_time` (earlier first, blank last), 4. `course_name` (A→Z), 5. `assignment_name` (A→Z). Ensure a setting or UI toggle switches between sort modes.	T04, T14
T17	**Inline difficulty editor** (1–5) on assignments. Persist to assignments.csv when storeData=true; update in-memory immediately; never overwritten by sync.	T03, T14
T18	**Auto-refresh scheduler**: 15-minute loop while app is open (pause/resume on focus).	T11, T12, T13
T19	**On-demand week fetch**: when navigating to a week not fetched this session, fetch once and merge; cache to CSV if storeData=true. Do not fetch data for anything in the past, once it's in the past.	T11, T12, T13
T20	**Settings screen** mirroring Orientation, plus **Change Token** and **Delete Token** actions. Writes to profile and updates runtime state without restart (if possible)	T01, T02, T06
T21	**Error banners & recovery**: 401/403 (pause and prompt for token), No Internet (show cached or empty with explanation), 429 (wait 60s then single retry), 5xx/timeout (treat as offline, recover next cycle). Centralize messages.	T10, T11, T18
T22	**Show refresh countdown** option: if enabled, render a visible countdown to next auto-refresh and reset after each run. Integrates with scheduler state. (BIG MAYBE IDK IF WE WANT TO DO THIS)	T18

T23	**Last viewed persistence**: save lastViewedWeekStart / lastViewedDay on navigation; restore on launch when openTo=last_viewed. Guard against stale/missing dates.	T01, T02, T04, T13

Sprint 3 (Complete by 11/05) [Week 10 & ½ Week 11]

Complete by 11/05. Meeting 11/05 @ 9:30 am

- **Amanda**: **T05**
- **Quentin**: **T06**
- **Kenny**: **T09**
- **Daniel**: **T12**

Definition of Done for Sprint 3

- Launch goes to correct flow (orientation already handled earlier) → token entry / continue offline works and returns to **Main**.
- Main has working tabs + week/day toggle + navigation; CSV-backed data can render when present.
- Merge correctly updates CSV by ID and returns “new/updated” list.

Sprint 4 (11/03- 11/14)
[Week 11 & Week 12]
Complete by: Meeting Monday 11/17 @ 4 pm

- **Quentin**: **T07**
- **Amanda**: **T14**
- **Daniel**: **T15**
- **Kenny**: **T19**

Definition of Done for Sprint 4

- Settings round-trip to profile and live-update app state where feasible.
- Sorting exists
- Scheduler runs every 15 minutes
- Visiting a new future/present week fetches once and merges; past weeks never refetched.

Sprint 5 (11/17-11/26)
[Goal to be completed by end of Week 13: Fri 11/21]
Complete by: Meeting (Tentative) Monday 11/24

- **Daniel**: **T21**
- **Amanda**: **Start on revising documentation**
- **Kenny**: **T23**
- **Quentin**: UI polish passes on banners, empty states, GUI in general. Help others if they need it.

Definition of Done for Sprint 5

- Error handling is user-visible, consistent, and non-blocking; one retry policy implemented.
  Optional countdown wired to scheduler state.
- App restores last viewed context when `openTo=last_viewed`.

Things to finish

1. List instructions for getting API token for orientation
2. Fix current day at the top oif the page, showing the correct day
3. Fix loading speed
4. Hook up created at field so you can view the bars correctly for how long you have to complete an assignment
5. Swap text for settings and viewing announcements tab
6. Make announcements actually display, only displays assignments right now
7. Change orientation tab name in the orientation to instructions
8. Get rid of settings stuff from token entry page
9. Change api token tab name to Token Login
10. Get rid of empty courses