

# Technical Requirements - Collapsible Chat Thread

Challenge is accessible on Memberspot: [https://pl-coding.mymemberspot.de/library/jx3b7Qik9ip5qpNI8IF2/s2aPahsnz9eGXhDmHaaB/FU\\_5JFvXVxjJnYLviKAhnz](https://pl-coding.mymemberspot.de/library/jx3b7Qik9ip5qpNI8IF2/s2aPahsnz9eGXhDmHaaB/FU_5JFvXVxjJnYLviKAhnz)

## Scenario

You're building a chat feature inspired by Reddit-style threads, where a single comment can spiral into a debate, then a side quest, then a code review. Users need a way to follow the conversation without getting buried. Your task: build a threaded UI where replies can nest infinitely, and tapping any comment neatly collapses or expands its entire branch — no scroll rage allowed.

## Figma Mockups

<https://www.figma.com/design/KqY7PRuFv4OP1i0aUTDFaM/Conversations?node-id=1-1029&p=f&t=vQPFo9nuULSTZJef-0>

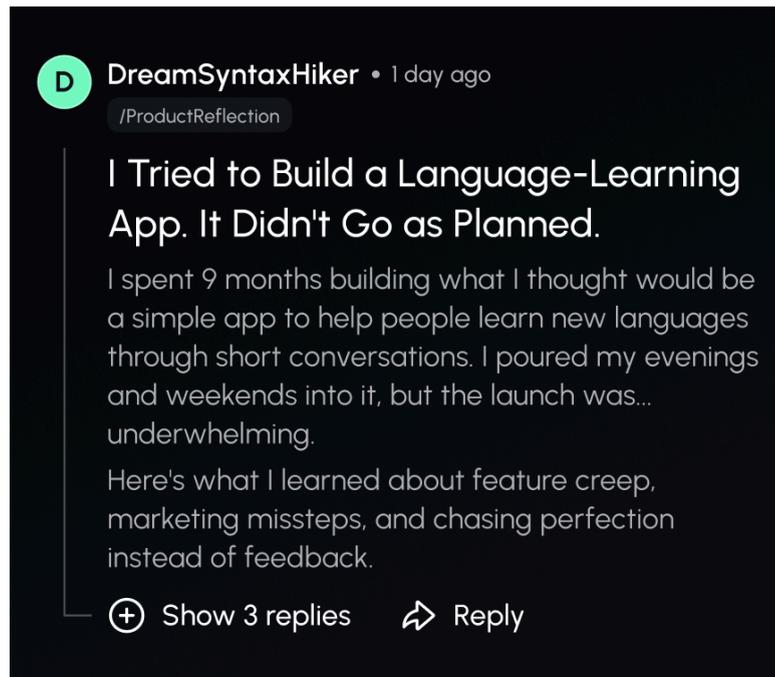
## Feature Goal

Build a threaded comment UI that supports nesting. Each comment shows the author, message, and timestamp, with a toggle to collapse or expand all nested replies beneath it. When collapsed, display a reply count for context. Focus on recursive data rendering, visual indentation, and clean state management for collapsing entire branches.

## Requirements

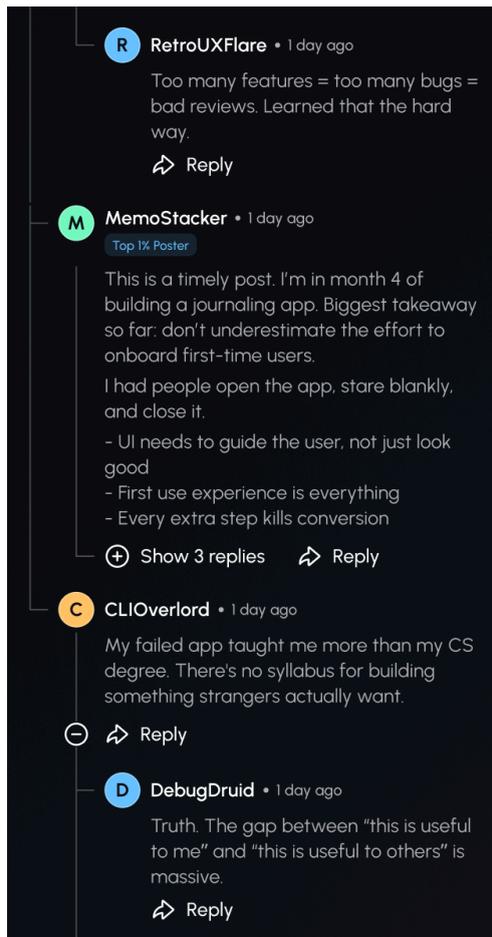
*Only dark theme is supported*

- The UI consists of a series of **Chat Message Components** organised in a vertical arrangement, with replies appearing nested.
- Each **Chat Message Component** displays:
  - The author's initial profile icon and author name
  - Relative date when the message was posted. Example: "1 day ago"
  - Chat channel text: "/ProductReflection"
    - This only applies to the original **Chat Message Component**
  - Title text
    - This only applies to the original **Chat Message Component**
  - Optional tags associated with the author
    - "Top 1% Commenter"
    - "Top 1% Poster"
  - The content of the message
  - Reply icon with text: "Reply"
    - *This icon has no click behaviour*



- Chat threads should use vertical and horizontal connector lines to represent nesting between messages visually. These lines help indicate the structure and flow of replies in the thread. The behaviour changes based on whether a comment has replies and whether it's expanded or collapsed.

Reply State	Vertical Line Below Icon	Action Icon/Text	Behavior
No replies	✗ None	✗ None	No lines or toggles shown.
Has replies (collapsed)	✓ Shown	+ "Show X replies"	Display number of hidden replies. On tap, expand the full nested branch.
Has replies (expanded)	✓ Shown	— Minus icon over the line	Show all nested replies. Tap to collapse the entire reply tree recursively.



## 🤔 What's Allowed?

- Standard Android/Jetpack libraries
- No 3rd party libraries are allowed or would be required to complete this challenge

## ⚠️ What's not important

- Responsiveness across every device size or orientation is not mandatory.

## 🧠 Disclaimer

The focal point of this challenge is the **expand and collapse logic** for threaded replies. Your goal is to manage recursive state cleanly and render nested threads intuitively. Don't worry about fancy animations, over-optimization, or full feature completeness — focus on getting the core interaction right.

## 🏆 Submission & Rewards

- A successful submission of this challenge via the `/submit-challenge` command on Discord grants you **150 XP**. You can use it in any channel on [Discord](#) :)
- A successful submission consists of these parts
  1. A link to a Gist showing your code for this challenge.
  2. A screen recording (40s max) showcasing:
    - a. Start with all thread collapsed
    - b. Expand each collapsed thread until they are all expanded

- c. Collapse the bottom-most thread
- d. Collapse the top-most thread