Technical Requirements – Thermometer Trek

Challenge is accessible on Memberspot: https://pl-coding.mymemberspot.io/library/jx3b7Qik9ip5qpNI8IF2/g98LzeMNxdeqdhGDiYMn/dJeukEmi0scUXQkbbbEU1/details

🎭 Scenario

This challenge simulates reading temperature data from a virtual sensor. Your goal is to build a UI that processes a data stream: filters it, transforms it, and displays only valid readings.

🎨 Figma Mockups

> https://www.figma.com/design/SAX4hR2rhXFUZgQjm1iZXF/Async-Adventures?node-id=0-1

🎯 Feature Goal

Display a stream of temperature readings in the UI, showing only valid values, and stop collecting data after 20 valid samples. After completion, the user should be able to start a new stream again.

📌 Requirements

- Light theme
- Centered layout
- Initial screen with a **Start** button
- After pressing **Start**, the UI switches to a list of readings and a progress indicator
- After the stream completes, a **Reset** button appears
- Pressing **Reset** starts a new stream — it does **not** return to the initial screen

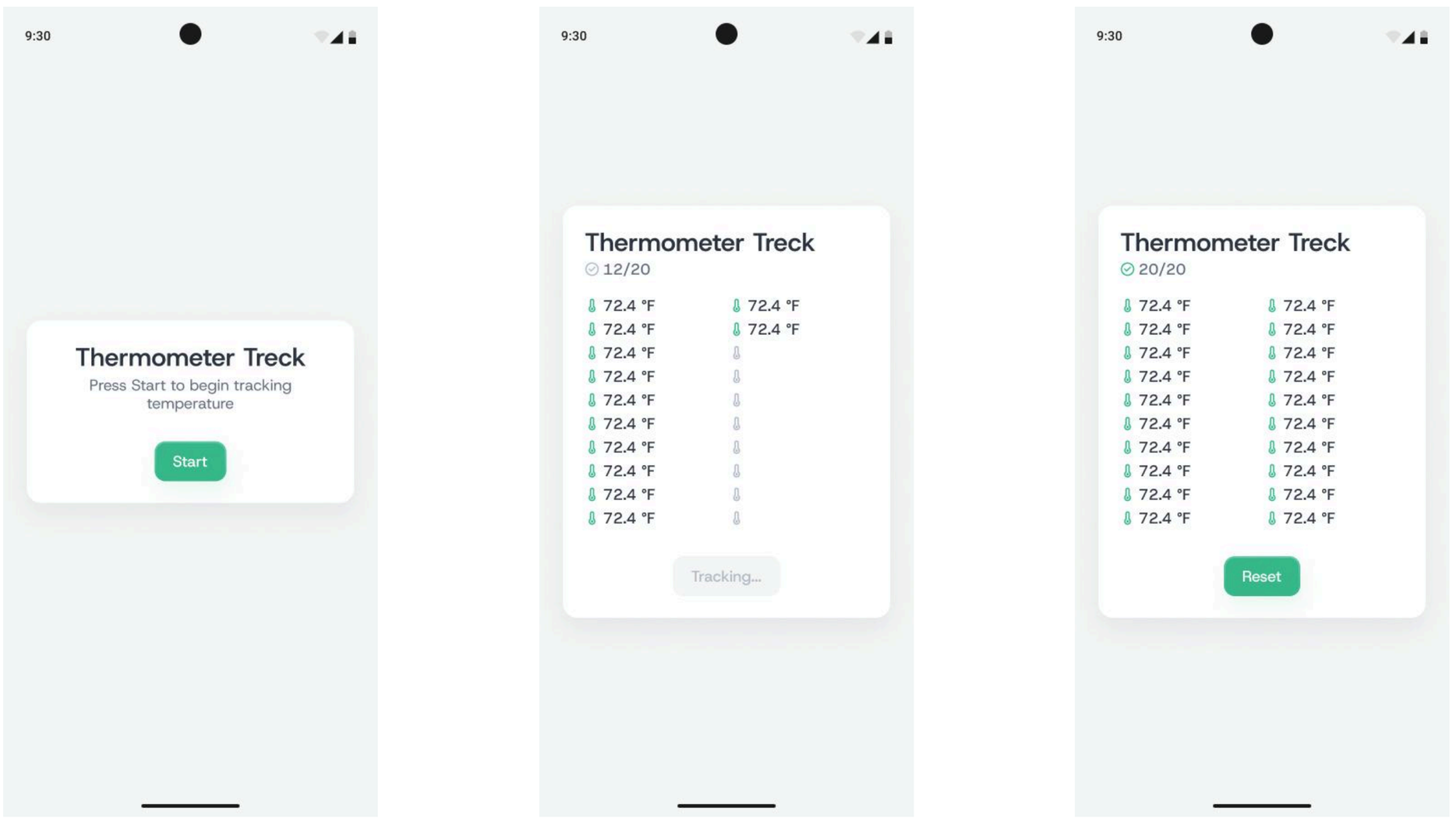
⚙️ How it works under the hood

- You are given a predefined list of temperature values that simulates the output from a virtual sensor:

```
listOf(
   -100.0, 0.0, -30.5, 10.0, 20.0, -60.0, 35.5, -51.3, 22.0, 19.8, 45.0, 55.1, 60.2,
   90.0, -49.9, 5.0, 12.3, 8.8, 0.5, -2.0, 30.0, 35.0, 27.5, 18.1, 15.6, 11.0, 17.3,
   33.8, 41.2, -80.0
)
```

Your task is to build a Flow that:

- Emits each temperature **every 250 ms**
- **Filters out** values **below -50 °C**
- **Converts** the remaining values to **°F** using the formula: °F = (°C × 9 / 5) + 32
- **Stops** after receiving exactly **20 valid samples**



🤔 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.
- Animations - although adding them would make a lot of sense 🙂

🏆 Submission & Rewards

- Successfully submitting this challenge via the /submit-challenge command on Discord grants you **100 XP**.
- Your submission must include:
  a. A link to a Gist with your implementation
  b. A screen recording (max 20 seconds) showing:
    - Starting the stream
    - Progressively adding temperature readings
    - Reaching 20 values
    - Restarting the stream via Reset