COMP 3550

2.2 — WRITING HIGH-QUALITY USER STORIES

Week 2: Planning Artifacts & Project Architecture

HOW TO IMPROVE A STORY

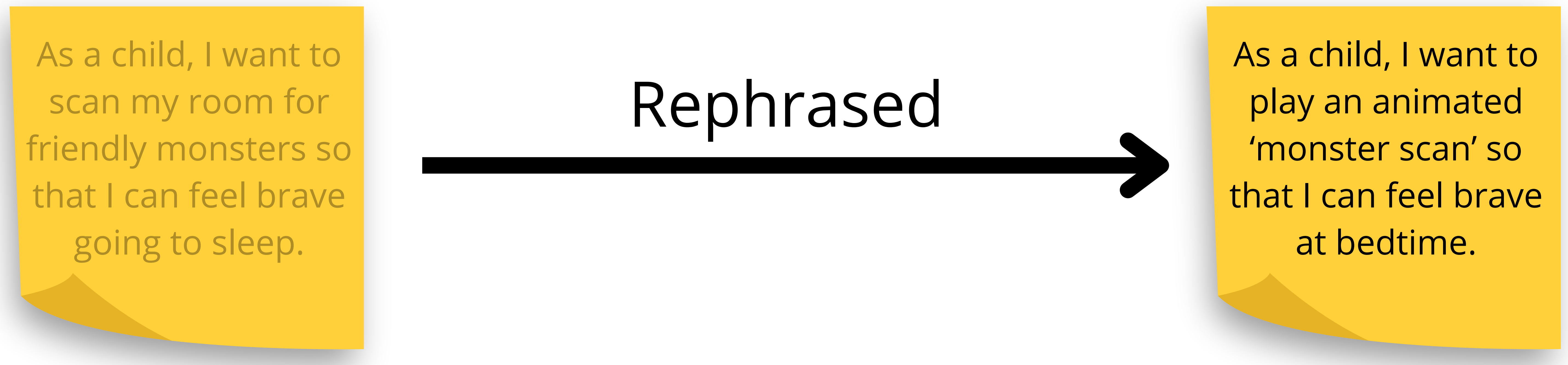
How do we actually improve a user story?

Let’s look at what we had before:

Independent?	✓	Yes — does not depend on badges, rewards, etc.
Negotiable?	✓	Yes — can discuss “scan” (camera? animation?)
Valuable?	✓	Yes — core emotional comfort feature.
Estimatable?	⚠	Depends on scan complexity (AR vs pretend scan).
Small?	⚠	Could get big if scope creeps (AR vision, voice)
Testable?	✓	Yes — can test child completes scan + feedback

As a child, I want to scan my room for friendly monsters so that I can feel brave going to sleep.

INVEST CRITERIA FOR GOOD STORIES



As a child, I want to scan my room for friendly monsters so that I can feel brave going to sleep.

Rephrased

As a child, I want to play an animated 'monster scan' so that I can feel brave at bedtime.

Estimatable?	✓	Now much easier to estimate — limited to an animation
Small?	✓	The story now scopes cleanly into a feature, undefined "scan" might have had scope creep

COMMON ANTI-PATTERNS

Watch out for these pitfalls!

Too big (“epics” pretending to be stories)
Break them down into smaller, testable chunks.

As a child, I want to have a complete bedtime adventure with scanning, badges, games, and stories so I can feel happy.

Technology-first wording
Implement API endpoint” ≠ user story! Focus on user value

Implement audio recording and playback.

No reason / no “so that”
Without a reason, it’s hard to prioritize or validate.

As a child, I want to scan for monsters.

Dependent on too many other stories
As the system, I want…” → no! Real users only.

As a child, I want to get a badge when I finish the checklist after playing the scan and hearing a message.

Fake users
If you can’t explain why it matters to a user, rewrite it.

As the system, I want to store messages in the cloud.

Unclear value
If you can’t explain why it matters to a user, rewrite it.

As a child, I want to customize monsters.

Well-written stories drive good design and good conversations — these anti-patterns get in the way!

PRIORITIZING BY VALUE

How do we get these values and how do we use them?

“As a parent, I want to record a custom goodnight message so that my child feels safe and connected at bedtime.”
PRIORITY: MEDIUM
TIME: 4 Days

“As a child, I want to scan my room for friendly monsters so that I can feel brave going to sleep.”
PRIORITY: HIGH
TIME: 1 Day

BREAKING DOWN STORIES

Our project artefacts start at Feature (not Epic), but other planning models have epics

Epic (5+ days) → **Feature** (3-5 days) → **Story** (1-3 days) → **Task** (<= 1 working day)

More on time estimations in 2.4

Monster Scan Experience

BREAKING DOWN STORIES

Our project artefacts start at Feature (not Epic), but other planning models have epics

Epic (5+ days) → **Feature** (3-5 days) → **Story** (1-3 days) → **Task** (<= 1 working day)

More on time estimations in 2.4

Monster Scan Experience

Feature ideas:

- Trigger scan interaction
- Play fun monster scan animation
- Show “room clear” message
- Play positive sound

BREAKING DOWN STORIES

Our project artefacts start at Feature (not Epic), but other planning models have epics

Epic (5+ days) → **Feature** (3-5 days) → **Story** (1-3 days) → **Task** (<= 1 working day)

More on time estimations in 2.4

Monster Scan Experience

Feature ideas:

- Trigger scan interaction
- Play fun monster scan animation
- Show “room clear” message
- Play positive sound

Example Stories:

- *“As a non-reading/small child, I want to start a monster scan so that I feel brave at bedtime.” (Trigger Scan)*
- *“As a child, I want to know the scan is happening so that the experience feels fun.” (Play fun monster scan animation)*

BREAKING DOWN STORIES

Our project artefacts start at Feature (not Epic), but other planning models have epics

Epic (5+ days) → **Feature** (3-5 days) → **Story** (1-3 days) → **Task** (<= 1 working day)

More on time estimations in 2.4

Monster Scan Experience

Feature ideas:

- Trigger scan interaction
- Play fun monster scan animation
- Show “room clear” message
- Play positive sound

Example Stories:

- *“As a non-reading/small child, I want to start a monster scan so that I feel brave at bedtime.” (Trigger Scan)*
- *“As a child, I want to know the scan is happening so that the experience feels fun.” (Play fun monster scan animation)*

Example Tasks (for Trigger Scan Story):

- Design “Start Scan” button
- Implement button click event
- Trigger animation on click
- Log scan event to system

OPTIONAL: ADD ACCEPTANCE CRITERIA

Make stories testable with clear conditions.

Format:

Given [starting context]
When [action happens]
Then [expected result]

<u>**Monster Scan Story**</u>

Given the app is open at bedtime,
When the child taps the “Start Monster Scan” button,
Then the scan animation plays and a “Room is safe” message appears.

Acceptance criteria = shared understanding → easier to test, easier to know when "done"!

BUILDING THE INITIAL BACKLOG

Start turning ideas into actionable stories.

So how do we do this?

Step 1: Create a shared document or board (Google Doc, Miro, Jira, Trello, etc.)

- We will be doing this in GitLab eventually

Step 2: Add your user stories — small, clear, user-focused (priorities & estimates)

Step 3: Link stories to architecture diagrams or planned modules (this would be REALLY great but I recognize the difficulty of it)

Step 4: Prioritize stories for MVP and early iterations

Step 5: Check INVEST — revise stories as needed

PAUSE & REFLECT

Revise one of your user stories using the INVEST framework.

- Add acceptance criteria if you can.
- What interface/system support does this story need?
- How long do you think it will take?