# Story Pack 2: Postmortem Gone Wild

## Scenario

Your product experienced a 2-hour outage. Slack logs and incident notes are available. Stakeholders want a structured postmortem, but no one has started it.

## Objective

Use LLMs to generate a blameless postmortem report and timeline of events.

## Step-by-Step Instructions

1. 1. Review provided logs and notes.
2. 2. Use the prompt:  
    'Write a blameless postmortem from these notes. Include timeline, impact, root cause, resolution, and next steps.'
3. 3. Critique the output: Is it factual, blameless, and complete?
4. 4. Assign roles and run the role-specific missions.
5. 5. Merge your work into a full team postmortem.
6. 6. Present how the LLM was used, what worked, and what didn’t.

## Common Pitfalls

* - Hallucinating causes or events
* - Assigning blame to individuals
* - Overly technical or vague reporting
* - Inconsistent tone

## Bonus Missions by Role

|  |  |
| --- | --- |
| Role | Challenge |
| Engineer | Extract accurate timeline of events from logs. |
| Architect | Prompt for root cause and long-term systemic fixes. |
| Product Owner | Summarize the business/user impact clearly. |
| Scrum Master | Generate talking points and team reflections for retro. |

## Pre-Task Checklist

☐ Have we defined the problem clearly?

☐ Have all team members picked a role?

☐ Have we run at least two prompt iterations?

☐ Have we validated the LLM output for accuracy and feasibility?

☐ Are we ready to present our solution back to the larger group?

## Team Notes

Team Brainstorm Notes:

Best Prompt We Used:

What We Improved After Feedback:

LLM Output Surprises:

## Team Reflection Questions

- What was your favorite LLM prompt or output?

- What didn’t go as expected during the activity?

- How did your team collaborate to solve the problem?

- Would you change your prompt strategy next time?

- What did this story teach you about LLMs and prompting?