Scenario 1

# Objective / Problem statement

1. The team lacks of purpose, also known as the “Why”. They are only fulfilling their tasks (the “What”). This is the root cause.
2. There is too much maintenance workload, this adds up to that the team feels like they are not building something meaningful.
3. The team members are not heard, which may lead to that they don’t want to contribute more than the assigned tasks.
4. They only [want to] know their own domain, leading to that they don’t want to raise knowledge gaps as impediments. They also may not feel safe to raise the impediments.
5. Late of story completion.

# Measurement for transparency

1. Sprint reviews and demos are on the development environment, without customers’ feedback.
2. 50:50 ration between maintenance and new features.
3. Ineffective dailies: people reporting own tasks, no impediments except for frustration (emotional impediments).
4. No acknowledgement for the emotional impediments, no retrospectives mentioned.
5. Burndown shows late stories closure. In addition, the team feels stressed while they still complete the stories, this may imply that there is hidden overtime.

# Solution to achieve / overcome

1. Facilitate retrospectives. A Mood Chart can be used in this case.
2. Highlight Transparency, Inspection and Adaptation, encourage raising impediments in the Dailies.
3. Replan the ratio between maintenance and new features. The maintenance tasks should be under 30%.
4. Introduce the technical debts and technical improvements tasks in the backlog. Set aside some budget each sprint for them (10% of the capacity).
5. Conduct knowledge sharing sessions to build cross-discipline members.
6. Check if the user stories were too big. Breaking them down into smaller parts to prevent end-of-sprint completion. The ideal stories should be around 3 days of work, and their sub tasks should be completable in 1 day. Regularly check the burndown for any symptom.
7. Involve customers to the reviews, focus on value delivered and get feedback from customers. Celebrate achievements and small wins.
8. Introduct regular project updates (could be monthly or every 2 months) so the team knows the value they have created, fostering ownership.

# Take Charge and Accountability

1. The Scrum Master to facilitate the ceremonies properly, particularly the Dailies, the Retrospectives, and the Reviews.
2. The Technical Lead to listen to the team, then design and manage the technical debts and improvements.
3. The Portfolio Manager to run the monthly project update.

# Constraints and risks

1. Rebalancing of maintenance may not be feasible due to the phase of the project, in this case, consider escalating to the management to use Managed Services.
2. Letting the team members speak up may lead to personal conflicts. Those sessions should be controlled carefully by the Scrum Master. Consider anonymous feedback if needed in the beginning.
3. Availability of the customers and stakeholders. In case they cannot join every review, at least they can see the video recordings and compliment or give feedbacks.

# Documentation and Learning

1. Everyone must have access to the bigger picture of the project, from the Solution to Release to Stories. This ensures that the team knows what they are contributing to while delivering their daily tasks.
2. The onboarding documents must be available, in which, there must be information about: The Customer, The Project, and The Team.
3. Centralized domain knowledge documents must be written by the according experts. Then hosting knowledge sharing sessions for the authors to present and Q&A. Everything must be planned as part of the sprint effort.

# AI Usage

For this scenario, I briefed MS Copilot the situation, then asked, each after one response from the AI:

1. What’s possibly happening here? Give me the analysis and reasons.
2. What are the potential limitations of the solutions?
3. What is the ideal ratio for maintenance?

# Conclusion (What I would tell the stakeholders?)

The team in this scenario is not agile, but they think that they are.

The team members have been trying to satisfy the stakeholders and customers by finishing their technical tasks. Technically, they can finish: the burndown chart reached the endpoint, however it was late in the sprint. In the end, the team doesn’t feel like accomplishing, while the customer doesn’t receive exactly what they are expecting, or maybe they don’t know since the team has not involve them.

They are now back on track, by bringing them back to the agile values: deliver small working products, regular communication, limit WIP, continuous improvement with cross-functional members.