

Group 4 – Approaches to a Sprint

Task: Read through the example approach for a Sprint below, use the [ladder of feedback](#) (source: Harvard Project Zero) to evaluate it and bring your own ideas into the discussion. One Scribe add notes below...

Tip: Try to picture using this approach with your community specifically.

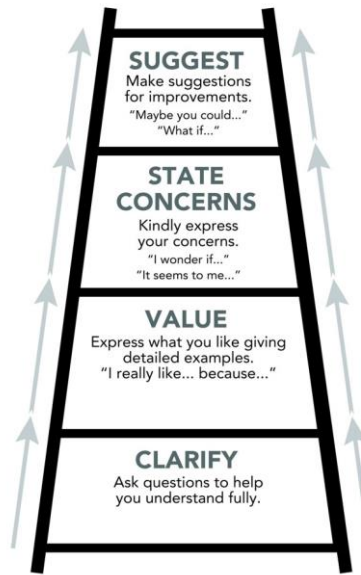
Group members: Heidi, Louie, Rebecca, Yuji, Arnett

Hackathon Approach:

- **Session 1: Problem Pitching and Team Formation**
 - Start with a "Problem Pitching" session where participants briefly present specific challenges or opportunities they face in their roles, related to the community's theme. After hearing the pitches, participants form small teams around the most compelling or relevant challenges, with each team choosing one problem to tackle during the Sprint.
- **Session 2: Rapid Ideation and Planning**
 - In this session, teams engage in "Rapid Ideation," brainstorming as many solutions as possible within a set time limit. Teams then evaluate their ideas and select the most viable one to develop. They spend the remainder of the session outlining a clear plan for how to bring their chosen idea to life, assigning roles and setting goals for what they want to achieve before the next session.
- **Session 3: Prototyping and Testing**
 - Teams focus on "Prototyping," creating a first draft or small-scale version of their solution. This could be a lesson plan, a strategy outline, or a tool, depending on the problem they're addressing. If time permits, they begin "Testing" by sharing their prototypes within the group or in their work contexts and gathering initial feedback to refine their approach.
- **Session 4: Presentation and Iteration**
 - The final session is for "Presentation," where each team showcases their solution to the broader group. They discuss what they created, how they tested it, and the results they observed. The group then engages in a "Feedback and Iteration" discussion, offering suggestions for further improvement. Teams conclude by outlining how they plan to continue developing or implementing their solutions after the Sprint ends.

Ladder of Feedback Notes on next page...

THE LADDER OF FEEDBACK



Ladder of Feedback

Suggest:

Scaffold the problem setting phase to make sure a good problem is chosen.

Could be good when you want to generate a tool e.g. in a learning support community.

The problem stated in the first sessions has to be well stated so that the solutions and recommendations are valuable.

Taking information from the survey may be very useful.

Be intentional about which members work together.

Source: [Harvard Project Zero](#)

Final reflection:

This is a great strategy, but it may not work well for some groups.

Needs to be something very concrete
State Concerns: People joining late would be an issue. Need to be really clear on what the problem is for this to work (so there's potential work if this isn't done well). The sprint might not be enough time for this process (not enough time for practicing and monitoring/evaluating impact). Might not be dynamic enough for a leadership community e.g. if something happens at school, the leader might want to talk about it (so it depends really on the purpose of the community) Concerned if people don't have the commitment to go back and try these things or are just turning up to sessions and that's it. Concerned if someone doesn't have the same problem and so is alone or that a different solution is needed for each place. Might end up feeling you're just helping someone else doing something meaningful for you.
Value: Continuity – everything is connected and flows. Like that there is one process and you are thinking deeply about one big idea. Potential for sense of achievement and creation of things of value Forces collaboration between people because they've joined in to solving each other's problems together (this might prevent absenteeism as well). You're getting feedback from people, suggestions for resources, improvements etc.
Clarify: Start here What type of community might this be useful for? How many people are needed at a minimum for the hackathon approach? What does the prototyping and testing look like? Are they sharing to the larger group or do they test? Are people going to be able to address problems in the sessions? What are people doing in between sessions? Will people be able to come up with all the issues/problems themselves? What role do you have in prompting achievable ideas? What happens if the process isn't working? What happens if people miss sessions? What happens if someone joins the community late?