

CSC 171 – Module 02

Create Agile Team

BUILDING THE CROSS-FUNCTIONAL, COLLABORATIVE TEAM

Team Requirements

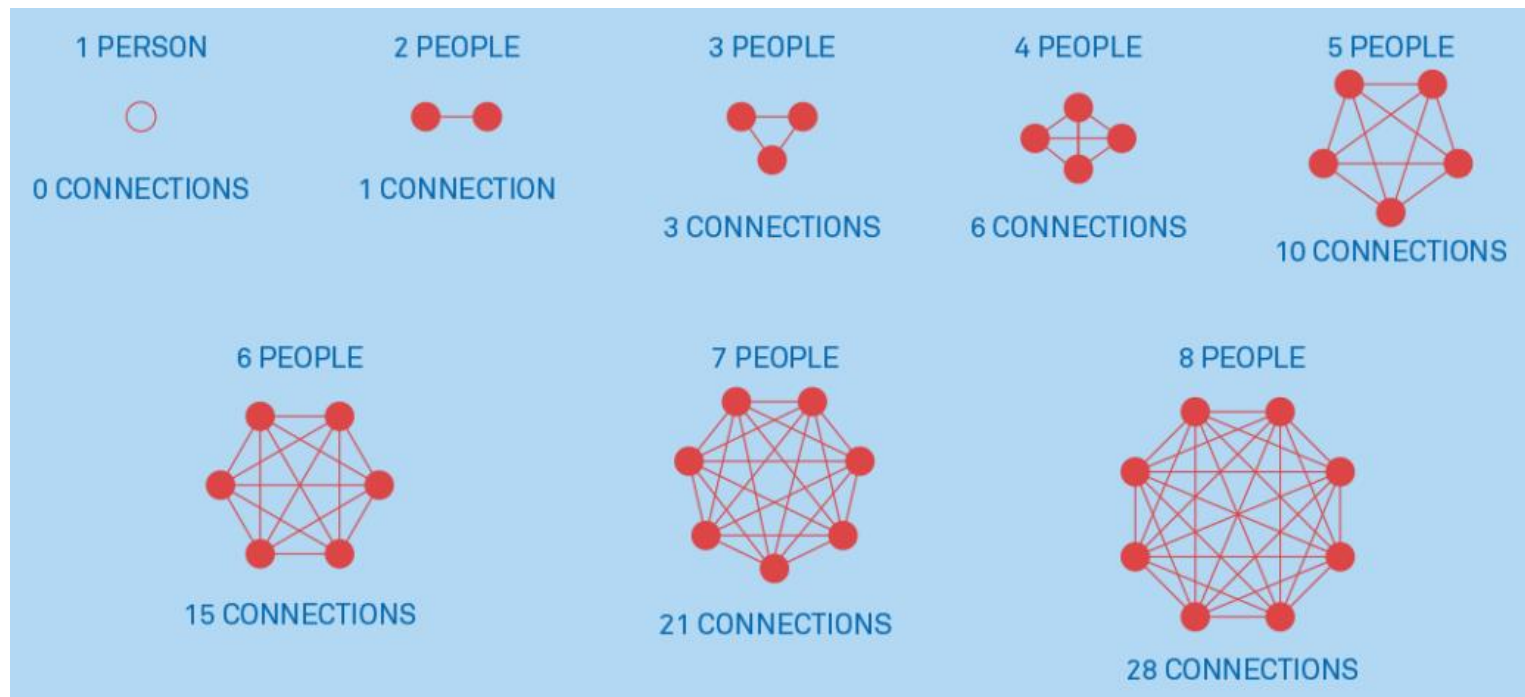
- The team has all the people (with skills and capabilities) it needs to complete the work
- The team has a shared goal for its project
- The team “owns” its work
 - Members commit to their work and they own their artifacts, including the code and test
- The team is stable, so they can learn to work together and learn
 - The team does not change the people inside an iteration
 - Team does not change people for a given feature

Team Roles

- Needs a full-time customer representative (product owner in Scrum)
 - Interacts with customers and stakeholders to create and rank the backlog
 - Decides which features the team will work on
- Development team
 - Makes its own work decisions and owns them
 - Decides how the team will do the work
- Might benefit from an agile coach
 - Help the team and organization in development agile culture
- Note that some agile methods (such as Scrum) have defined roles while others (such as Kanban) do not have specific roles

Team Size

- ≤ 3 , might not have enough diversity to solve their problems
- > 9 , too many communication paths



Team Organize Themselves Example

- Explain different feature sets (or products) that are needed
- Ask members to select which feature set (or product) they like to work on
- Record member names and their capabilities for each feature set (or product)
- Identify capabilities that might be missing
- Team can add missing capabilities through learning or hiring people with required capabilities

Identify Team's Values

- Values are how people treat each other
- An activity
 - Ask everyone to meet for about 30 minutes
 - Each member has an index card and a magic marker
 - Each member fills in the sentence:
 - “I don’t like it when someone/people ...”
 - Write down anywhere from 2 to 5 of these sentences
 - Divide the team into pairs
 - Each pair selects one card...
 - Write down a statement that counters the negative statement
 - Continue until each pair addresses all the statements
 - Have the pairs read each “like it when...” statement
 - Capture the counter statements (values)
- Example
 - Statement
 - “I don’t like it when some people tell me what to do”
 - Counter statement
 - “I like it when people discuss our technical approach as a team”

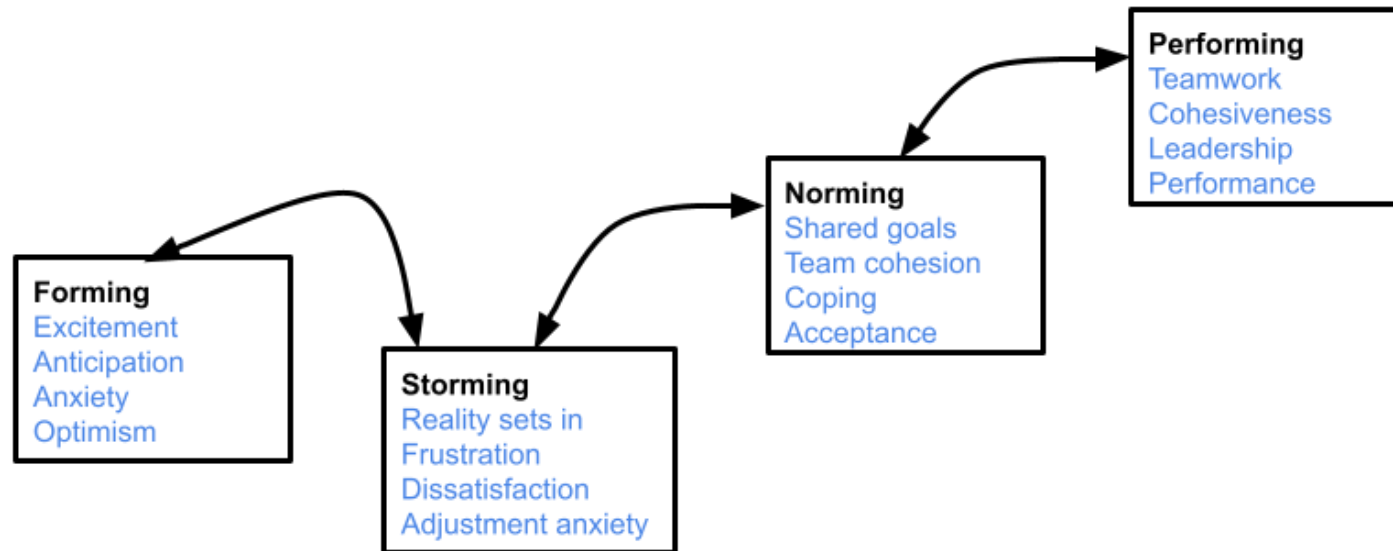
Develop Working Agreements

- Working agreements define the ways team members work together, include
 - What "done" means
 - e.g., Is creating tests for new code part of "done"?
 - Ground rules of meetings
 - Team norms
 - e.g., core hours
 - ...
- Working agreements are not static

Types of Teams

| Who is Responsible For | Manager-Led Teams | Self-Directed Teams | Self-Managing Teams | Self-Governing Teams |
|---|-------------------|----------------------------------|---------------------|----------------------|
| Setting the overall direction | Manager | Manager | Manager | Team |
| Designing the team & its organizational context | Manager | Manager | Team | Team |
| Monitoring & managing work process and progress | Manager | Team, except for hiring & firing | Team | Team |
| Executing team tasks | Team | Team | Team | Team |

Tuckman's Model of Group Development



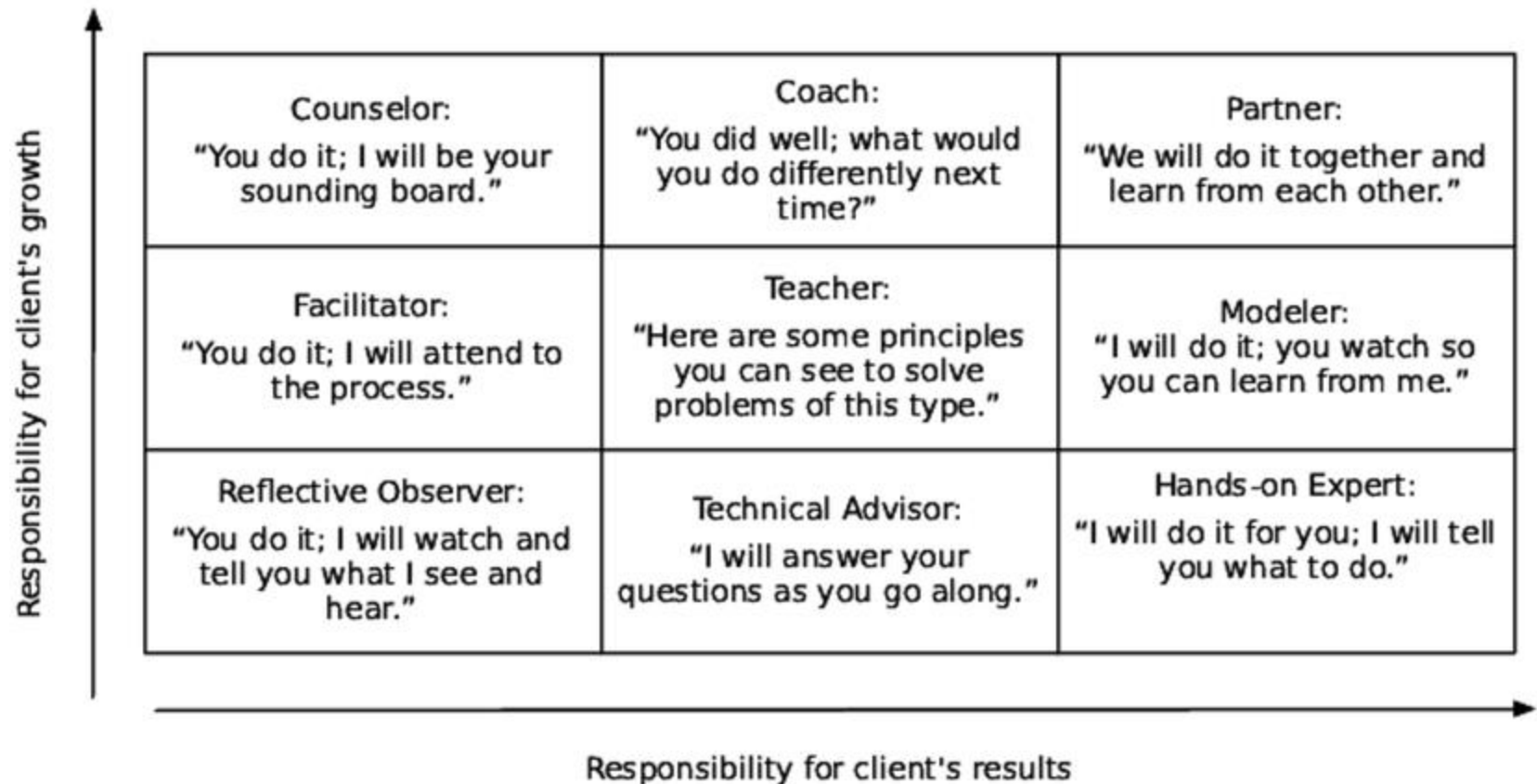
Because a team takes a while to learn to work together, keep it together.

BUILD TEAMWORK WITH INTERPERSONAL PRACTICES

Team & Interpersonal Skills

- Agile team need interpersonal skills
 - Communication skills, emotional intelligence, team-working, negotiation, persuasion, and influencing skills, conflict resolution and mediation, problem solving and decision making
- Agile team members
 - Can receive and provide feedback
 - Change-focused and reinforcing feedback
 - Can coach each other
 - Can collaborate with each other
 - Can ask each other for help
 - Are adaptable, willing to work on whatever is next and possibly outside of their expertise
- Agile team members
 - Do something good enough for now (as opposed to waiting for perfection)
 - Create experiments to try something and receive feedback on the product or the team's process
 - Be willing to work outside their preferences as a generalizing specialist

Possible Consulting Roles [1]



Building Trust

- Only happens when the team work together
- Team members are trustworthy when they do the following
 - Deliver what they promise to deliver
 - Are consistent in their actions and reactions
 - Make integrity a cornerstone of their work
 - Are willing to discuss, influence, and negotiate
 - Trust in themselves and their team members

Creating an Environment of Safety

- Safety
 - Allows the team to manage the ambiguity and uncertainty about anything related to the work
 - Allows the team to learn early by creating small experiments
- Safety creates an environment in which team members feel safe
 - Helping team members admit and look for mistakes
 - Making possible team and team member learning
 - Facilitating the use of clear and direct language by all team members
 - Providing an environment where “admitting what you don’t know” and “acknowledging when you fail” is expected

| | Low Accountability | High Accountability |
|-------------|--------------------|----------------------------------|
| High Safety | Comfort Zone | Learning & High Performance Zone |
| Low Safety | Apathy Zone | Anxiety Zone |

Anti-Collaboration Patterns

- When the team's **status** is uneven, members are reluctant to take a risk in front of managers
- When the team feels **uncertain** about its next steps
 - e.g., undertakes a large, not well understood Feature
- When the team has insufficient **autonomy** to work the way they want, the team is bound by decisions outside the team
- When team doesn't know or can't manage its team membership (**relatedness**), the team does not have the ability to develop member relationships
- When team members sense **unfairness** in how they are treated

SCARF: **S**tatus, **C**ertainty, **A**utonomy, **R**elatedness, **F**airness

AGILE REQUIRES DIFFERENT PROJECT LEADERSHIP

Traditional Project Manager

- Traditional project manager is responsible for [2]
 - Developing the project management plan and all related component plans
 - Keeping the project on track in terms of schedule and budget
 - Identifying, monitoring, and responding to risk
 - Providing accurate and timely reporting of project metrics
- Scrum
 - Replaces the project manager role with ScrumMaster
 - Splits the project manager's responsibilities into
 - ScrumMaster
 - Product Owner
 - Development team

Servant Leader

- A servant leader in the form of an agile project manager or a coach might be needed to help the team understand the process it needs.
- Servant leader practices
 - They are self-aware.
 - They listen.
 - They serve the people who work “for” them.
 - They help other people grow.
 - They coach people, not control them.
 - They unleash the energy and intelligence of others.
 - They work to develop their foresight so they can act, not react.

Agile Project Manager

- Is responsible for developing the team and instilling agile values and practices
- Plays the role of a facilitator to ensure all the roadblocks are removed
- Does
 - Facilitate the team's process (e.g. Scrum)
 - Remove "impediments" that are negatively affecting the team's work
 - Assist the team in measuring the team's velocity, cycle time, and other measurements
 - Assist the team in working with the Product Owner in writing user stories
 - Facilitate the team's discussion and activity in developing the Project's Vision, the Release Criteria, the definition of "done", etc.
 - Help in identifying and managing the team's risks
 - Allowing team to set up time for learning
 - Ensuring the team does not have too much context switching
 - Obtaining more resources (including funding) if necessary
 - Making sure that long lead-time items show up on time
 - ...
- Does not do
 - Assign work
 - Estimate work on behalf of the team
 - Commit to features, stories, or tasks on behalf of the team
 - Agree to dates
 - Agree to constraints on the project

Product Owner

- Represents the customer to the team
- Creates the Product Backlog
 - Identifies the deliverables, the features, and the user stories associated with each feature
 - Prioritizes the order in which the work will be done
- Manages the Product Backlog
 - Clearly expressing Product Backlog items
 - Ordering the items in the Product Backlog to best achieve goals and missions
 - Optimizing the value of the work the Development Team performs
 - Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the team will work on next
 - Ensuring the Development Team understands items in the Product Backlog to the level needed
- Decides what to do and when the team should deliver that work

References

- [1] Create Your Successful Agile Project, Johanna Rothman, Pragmatic Programmers LLC, 2017. ISBN:9781680502602
- [2] Sutherland, Jeff and Nafis Ahmad August. “How a Traditional Project Manager Transforms to Scrum –.” (2011).