

CEN 5016: Software Engineering

Fall 2026



University of
Central Florida

Dr. Kevin Moran

Week 3 - Class 11: Software Teams & Communication Pt. I





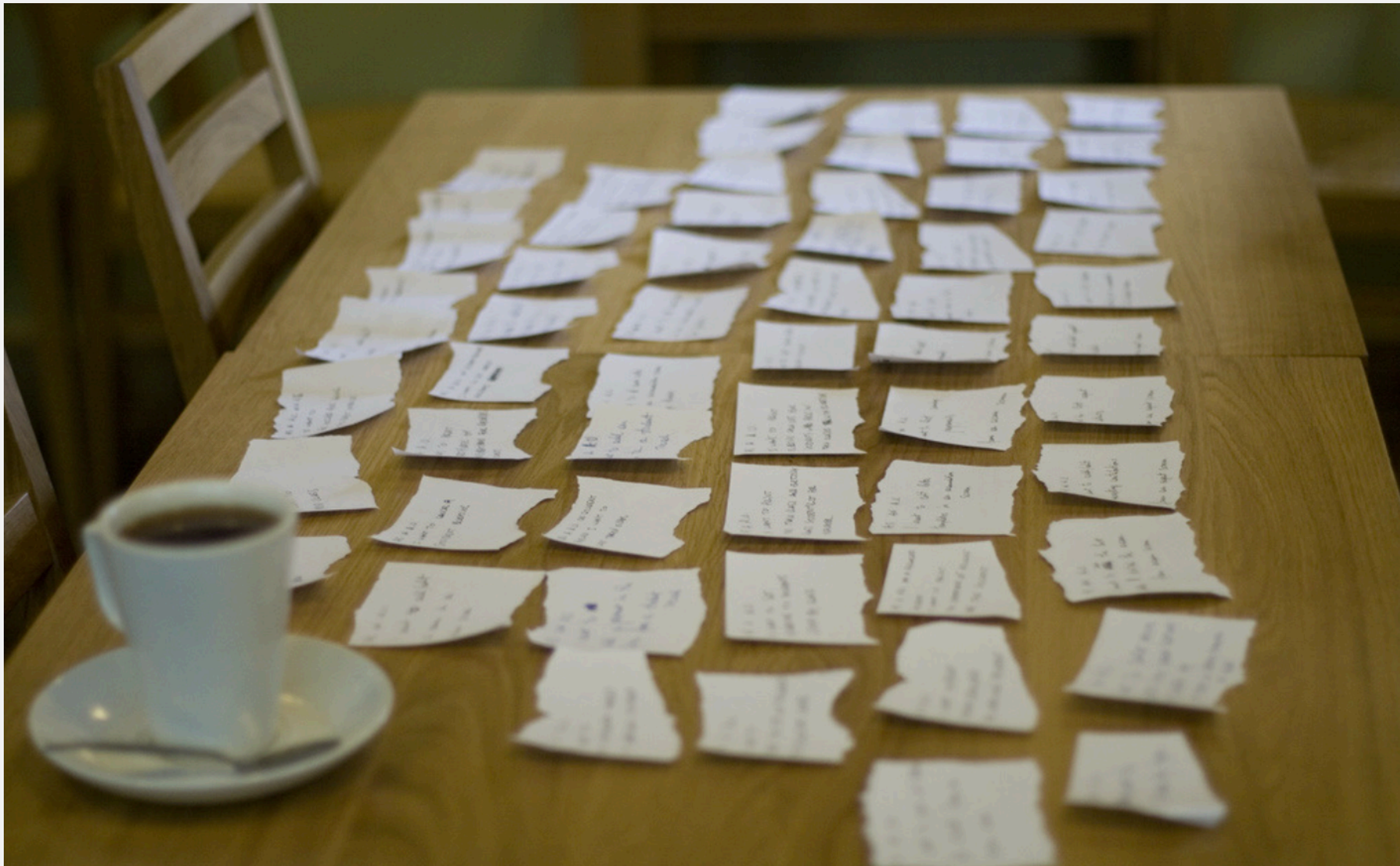
- *Team-forming this week - Due Friday, Sept 11th EoD!!*
 - Teams of 3-4 students
 - See Ed Discussions Post
- *Assignment 2 + SDE Project Posted!*
 - See post on Ed and Assignment page on Course Website
- *Dr. Moran will be out next week (Attending ICSME'26)*
 - My PhD student Atish Dipongkor will be standing in for lectures (he is great!)



Project Planning & Agile Development



User Stories



User Stories



card

a brief, simple requirement statement from the perspective of the user

conversation

a story is an invitation for a conversation

confirmation

each story should have acceptance criteria

one | 80

SEITE 80



- “As a [role], I want [function], so that [value]”



- What must a developer do to implement this user story?



- How can we tell that the user story has been achieved
- It's easy to tell when the developer finished the code.
- But, how do you tell that the customer is happy?

How to Evaluate a User Story



Follow the INVEST
guidelines for good
user stories!



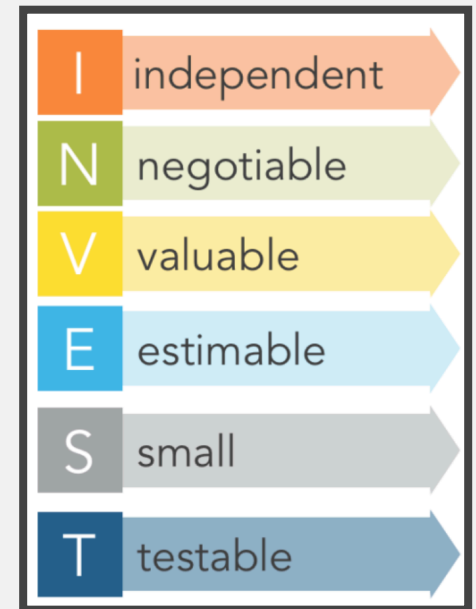
one | 80
SERVICES



Independent



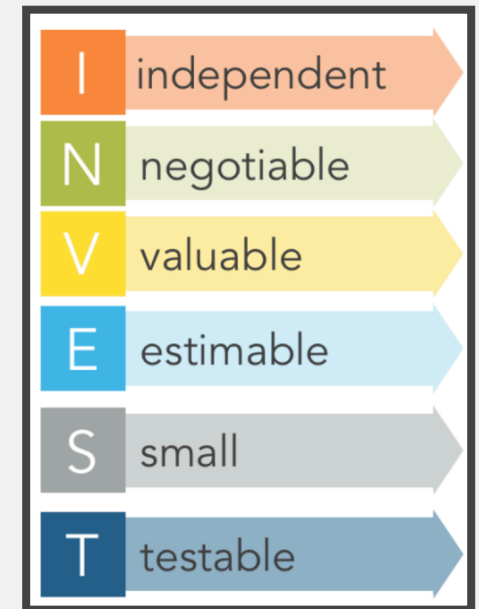
- Schedule in any order.
- Not overlapping in concept.
- Not always possible.



Negotiable



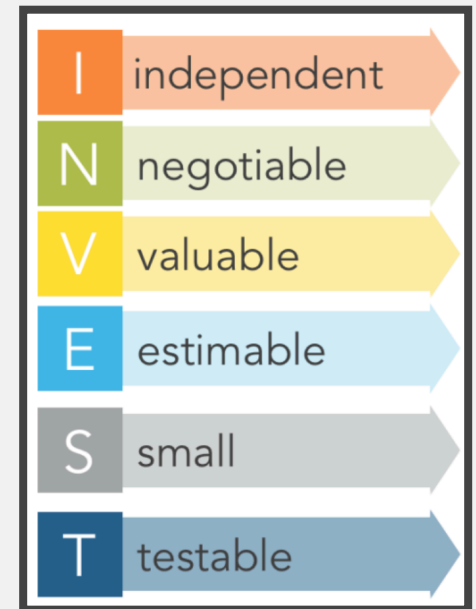
- Details to be negotiated during development.
- A good story captures the essence, not the details.



Valuable



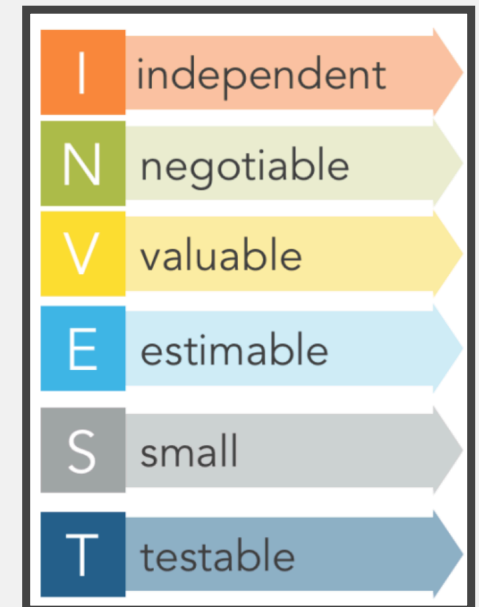
- This story needs to have value to someone (hopefully the customer).
- Especially relevant to splitting up issues.



Estimable



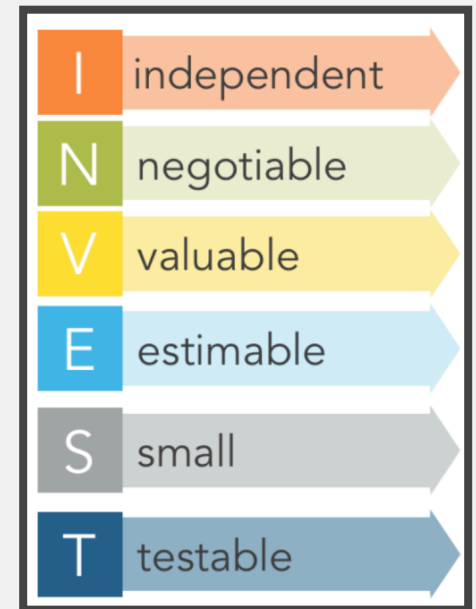
- Helps keep the size small.
- Ensure we negotiated correctly.
- “Plans are nothing, planning is everything” - Dwight D. Eisenhower



Small



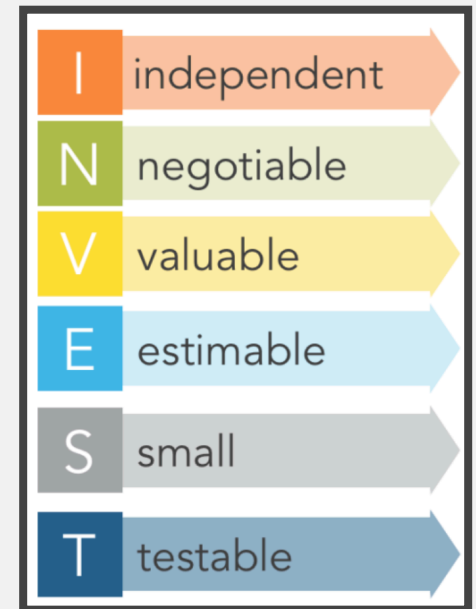
- Can be written on a 3x5 card.
- At most two person-weeks of work.
- Too big === unable to estimate



Testable



- Ensures understanding of task
- We know when we can mark task “Done”
- Unable to test === I do not understand it



Software Teams & Communication



Learning Goals



- Describe the pros and cons of working as a team
- Recognize the importance of communication in collaboration
- Recognize the need of having multiple communication channels
- Select an appropriate communication tool for a given communication goal
- Ask technical questions effectively
- Write clear and specific Github issues, pull requests, and comments

We all Work in a Team



Bubble Sort

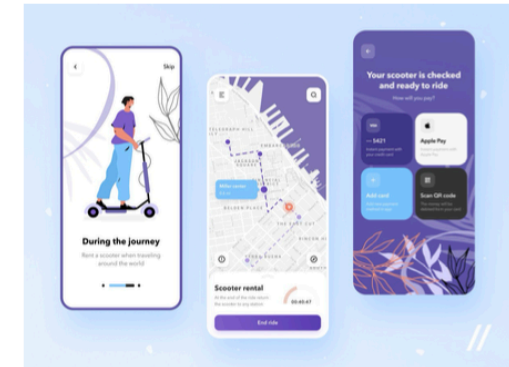
First pass	6	2	8	4	10
Next pass	2	6	8	4	10
Next pass	2	6	4	8	10
	2	4	6	8	10

Review complete

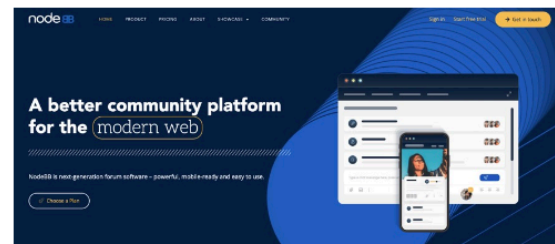
Bubble Sort



Monopoly Game



Scooter App

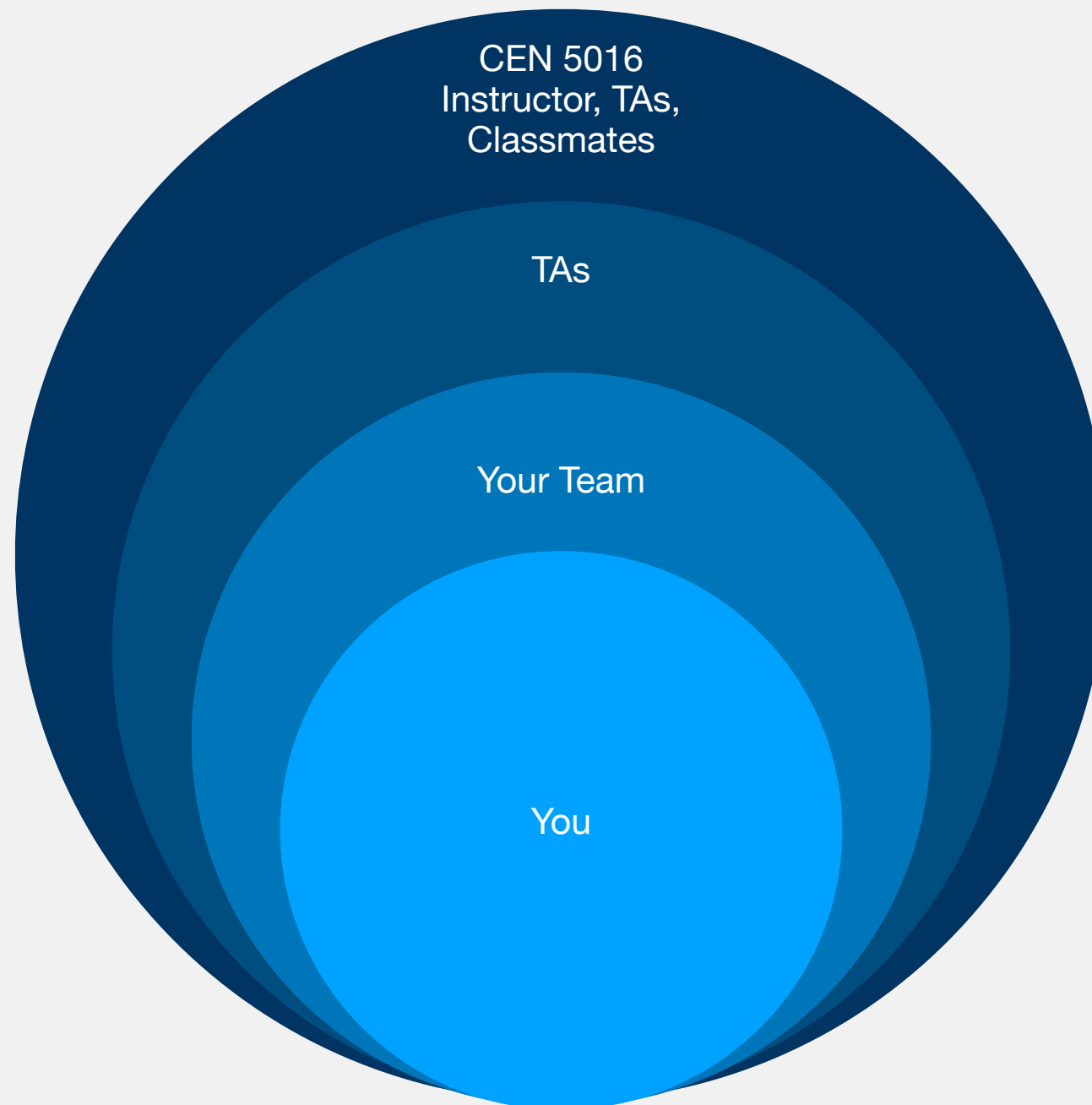


NodeBB



Autonomous Vehicle

We all Work in a Team



Activity: Working Solo vs. As a Team



- Write down two pros and two cons about working solo on a software project versus working on a team.
- Pair with your neighbor and discuss your answers. Do you agree?
- We will discuss some of your answers!

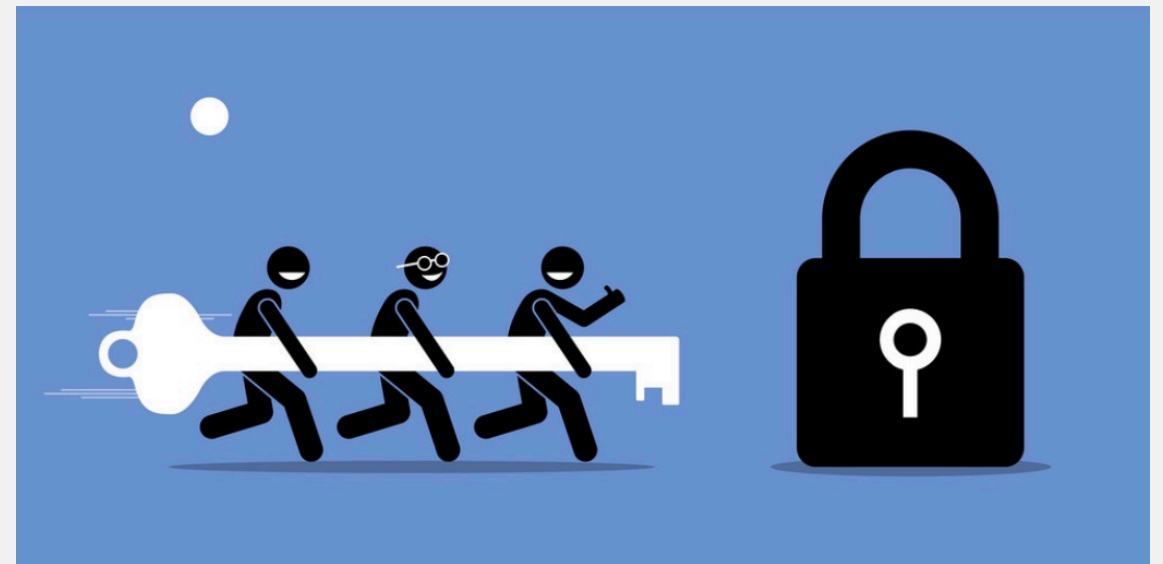


<https://tinyurl.com/CEN5016-ACT3-F26>

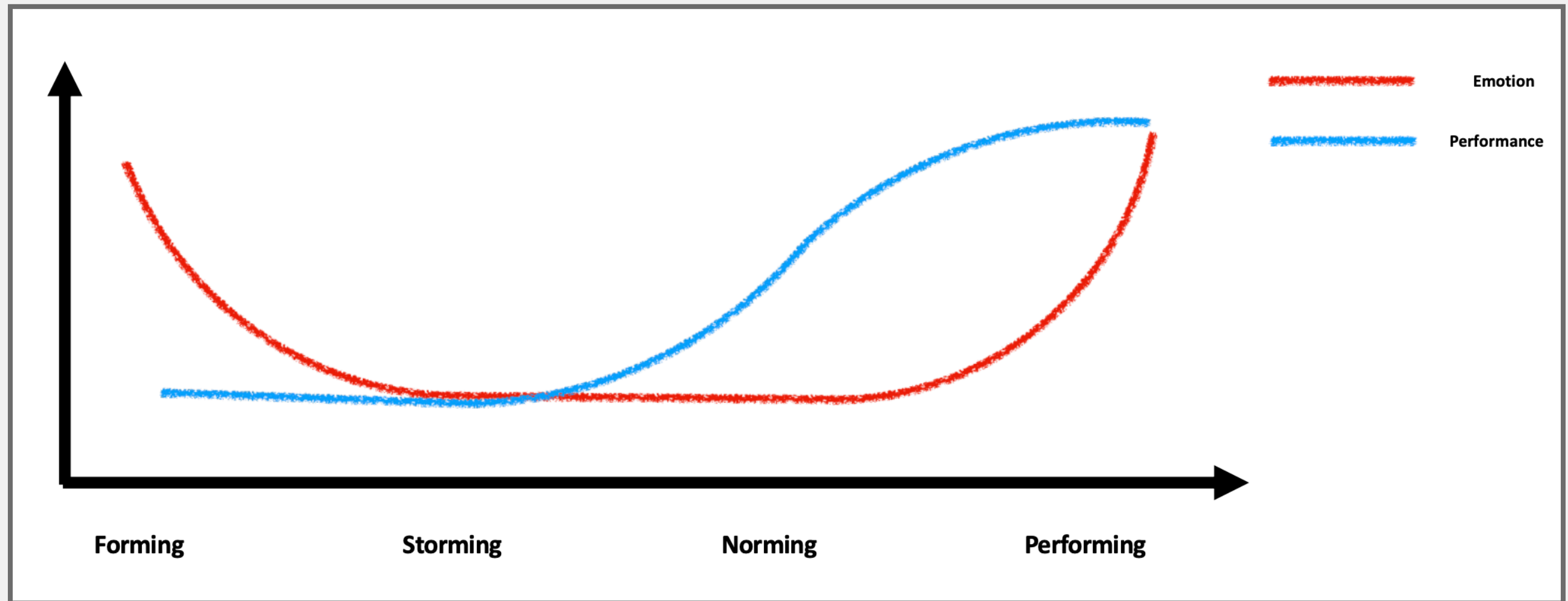
Working as a Team



- Design & implement software
- Establish a collaboration process
- Meet with the team
- Choose a leader
- Divide work and integrate
- Share knowledge
- Resolve conflicts



Stages of Team Formation





- When working with someone who is remote, how do you like to work together?
- How do you manage your time when you get busy with a lot of tasks?
- How do you feel about chatting by text message, audio call, video call?
 - Exchange phone numbers with your project partner(s) in case your Internet goes out and you still want to work on the project together.
- Negotiate when you can work on the project together outside of class.
- Have you had a positive prior teaming experience?
 - How often did your team meet?
 - Did your team have a leader? If yes, what did that leader do?
 - What was your role on the team?
 - How well did you get along with your teammates related to work, or related to non-work?



SIMPLE SABOTAGE
FIELD MANUAL
Strategic Services
(Provisional)

What Not to Do 🤔



(11) General Interference in Production

(a) Organizations

(1) Insist on "channels." Never in order to expedite.

(2) Make "sp" possible and at "points" by long personal experiences. appropriate "pat" can.

(3) When p committees, for "tion." Attempt to as possible — nev

(4) Bring up as possible.

(5) Haggle communications, min

(6) Refer ba the last meeting question of the a

(7) Advocate and urge your f able" and avoid embarrassments

(8) Be wor decision — raise action as is cont diction of the gro with the policy o

(b) Managers and Supervisors

(1) Demand written order

(2) "Misunderstand" orde questions or engage in long about such orders. Quibble over can.

(3) Do everything possibl delivery of orders. Even though may be ready beforehand, don't it is completely ready.

(4) Don't order new wo until your current stocks have bhausted, so that the slightest your order will mean a shutdo

(5) Order high-quality ma hard to get. If you don't get th it. Warn that inferior materia ferior work.

(6) In making work assi sign out the unimportant job the important jobs are assign workers of poor machines.

(7) Insist on perfect work important products; send back those which have the least flav defective parts whose flaws a the naked eye.

(8) Make mistakes in rout and materials will be sent to th

(12) Multiply paper work in plausible ways. Start duplicate files.

(13) Multiply the procedur involved in issuing instructions so on. See that three people everything where one would do

(14) Apply all regulations

(c) Office Workers

(1) Make mistakes in qua when you are copying orders names. Use wrong addresses.

(2) Prolong corresponder ment bureaus.

(3) Misfile essential docu

(4) In making carbon cop few, so that an extra copying be done.

(5) Tell important caller or talking on another teleph

(6) Hold up mail until t

(7) Spread disturbing r like inside dope.

(d) Employees

(1) Work slowly. Think crease the number of movem your job: use a light hammer one, try to make a small wren one is necessary, use little for able force is needed, and so o

(2) Contrive as many int work as you can: when char on which you are working, a lathe or punch, take needles

(3) Even if you understand the language pretend not to understand instructions in a foreign tongue.

(4) Pretend that instructions are hard to understand, and ask to have them repeated more than once. Or pretend that you are particularly anxious to do your work, and pester the foreman with unnecessary questions.

(5) Do your work poorly and blame it on bad tools, machinery, or equipment. Complain that these things are preventing you from doing your job right.

(6) Never pass on your skill and experience to a new or less skillful worker.

(7) Snarl up administration in every possible way. Fill out forms illegibly so that they will have to be done over; make mistakes or omit requested information in forms.

(8) If possible, join or help organize a group for presenting employee problems to the management. See that the procedures adopted are as inconvenient as possible for the management, involving the presence of a large number of employees at each presentation, entailing more than one meeting for each grievance, bringing up problems which are largely imaginary, and so on.

(9) Misroute materials.

(10) Mix good parts with unusable scrap and rejected parts.

General Devices for Lowering Morale and Creating Confusion

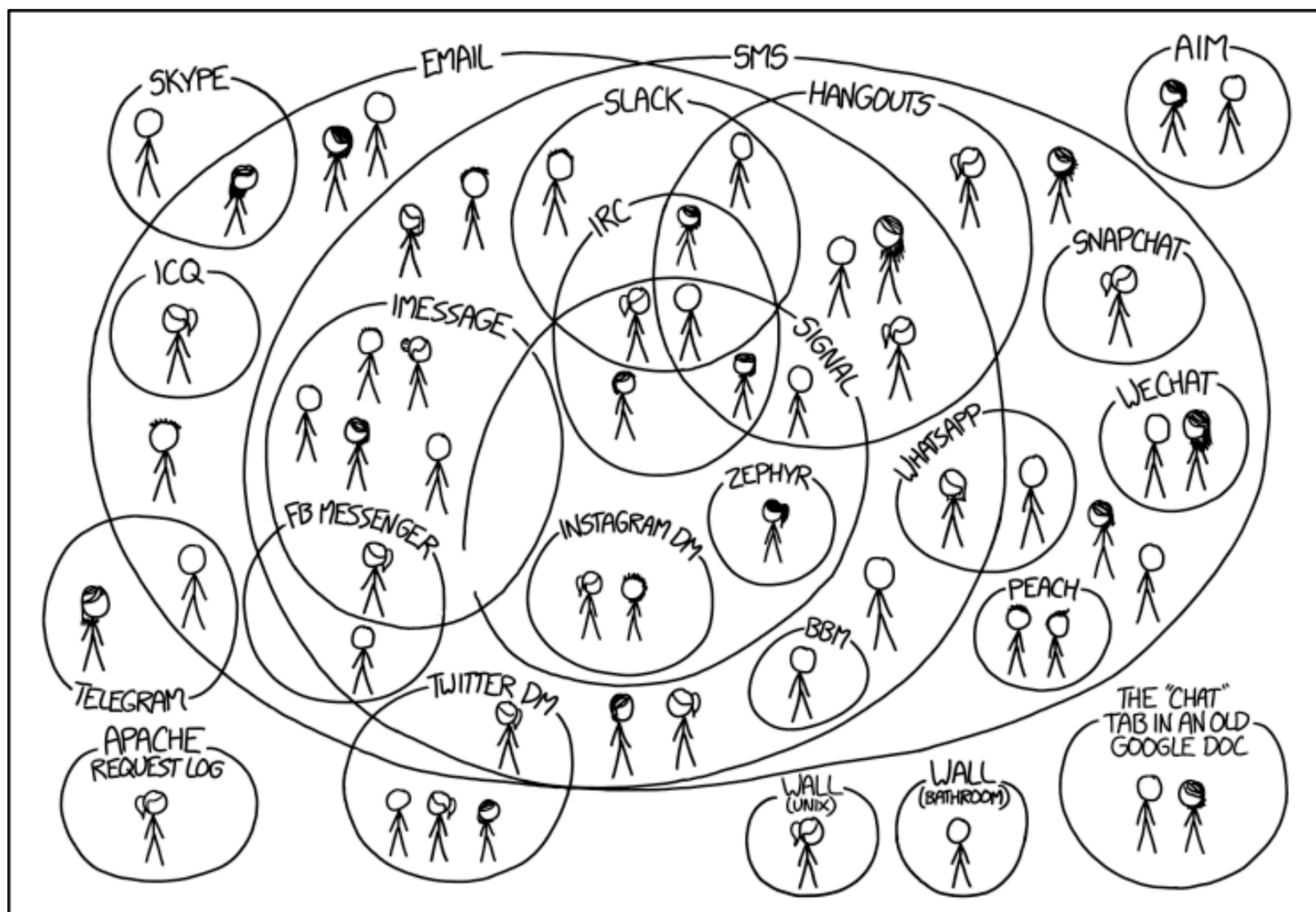
(a) Give lengthy and incomprehensible explanations when questioned.

(b) Report imaginary spies or danger to the state or police.

Establish a Collaboration Process

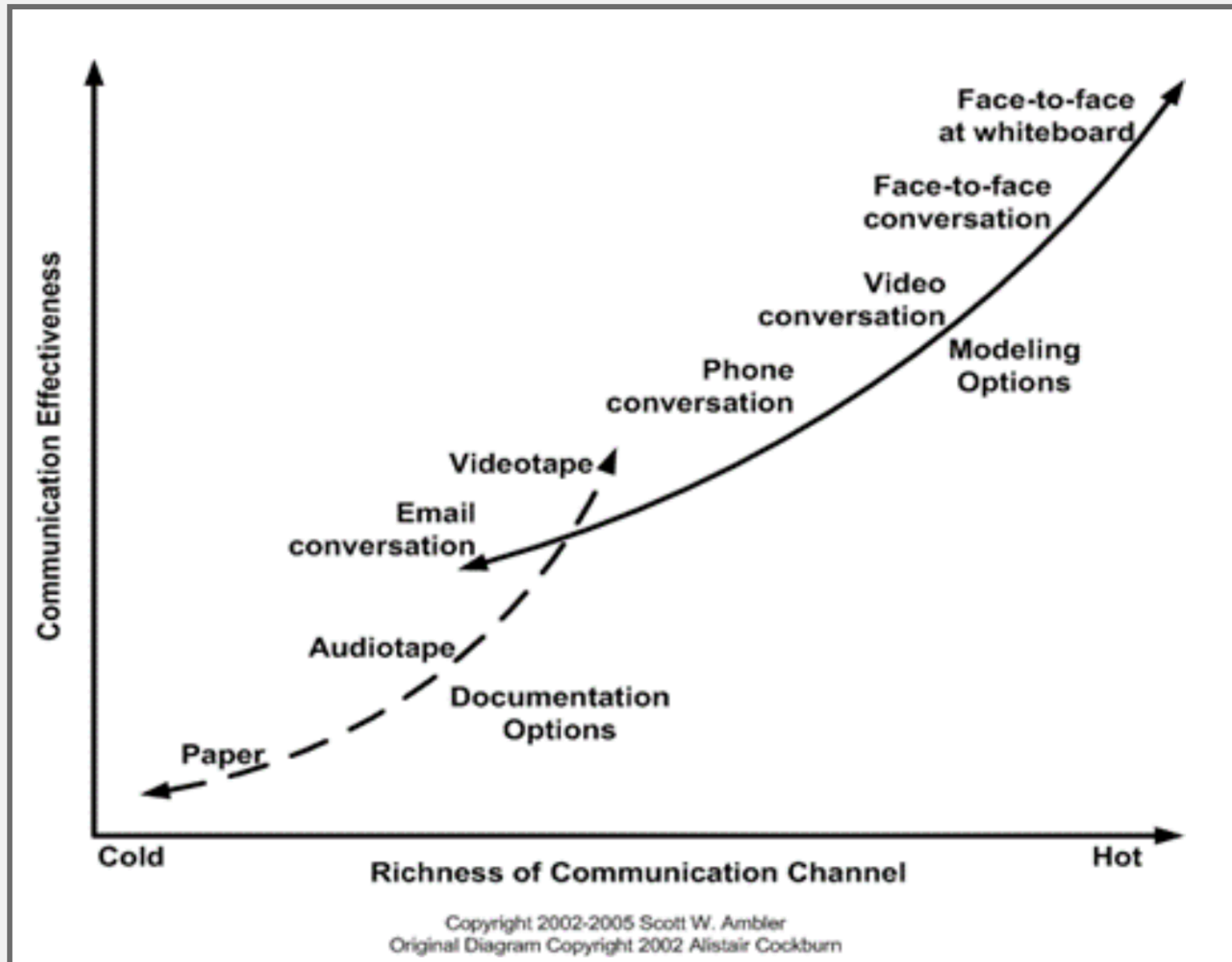


Communication App Confusion



I HAVE A HARD TIME KEEPING TRACK OF WHICH CONTACTS USE WHICH CHAT SYSTEMS.

Select the Right Communication Tools



Establish Communication Patterns



- Asana, Trello, Microsoft Projects, ...
- Github Wiki, Google Docs, Notion, ...
- Github Issues, Jira, ...
- Email, Slack, Facebook groups, ...
- Zoom, Microsoft Teams, Skype, Phone call, ...
- Face-to-face meetings



- Ed Discussions
- Regular meeting (Lectures, Recitations)
- Office Hours
- Webcourses
- Course Webpage



Communication

- Forums: Discuss implementations, research, etc. <https://discuss.pytorch.org>
- GitHub Issues: Bug reports, feature requests, install issues, RFCs, thoughts, etc.
- Slack: The [PyTorch Slack](#) hosts a primary audience of moderate to experienced PyTorch users and developers for general chat, online discussions, collaboration, etc. If you are a beginner looking for help, the primary medium is [PyTorch Forums](#). If you need a slack invite, please fill this form: <https://goo.gl/forms/PP1AGvNHpSaJP8to1>
- Newsletter: No-noise, a one-way email newsletter with important announcements about PyTorch. You can sign-up here: <https://eepurl.com/cbG0rv>
- Facebook Page: Important announcements about PyTorch. <https://www.facebook.com/pytorch>
- For brand guidelines, please visit our website at pytorch.org

Communication Expectation



- Quality of service guarantee
 - How soon will you get back to your teammates?
 - Weekend? Evening?
- Emergency
 - Tag w/ 911
 - Notify everyone with @channel

Running a Meeting



How to Run a Meeting



- The Three Rules of Running a Meeting
 - Set the Agenda
 - Start on Time. End on Time.
 - End with Action Items (and share them - Github Issues, Meeting Notes, ...)

How to Run a Meeting



- Set and document clear responsibilities and expectations
- Make everyone contribute
 - Possible Roles: Coordinator, Scribe, Checker
 - Manage Personalities
 - Be Vulnerable

Atlassian Meeting Flowchart



Every Team Needs a Leader & a Manager



- Note: these are not the same thing.
- A leader inspires with their vision of how everyone could work together.
 - They maintain a positive working environment.
 - They actively create their team culture.
 - They promote fair play among team members.
 - They acknowledge their team members' individuality.
 - They are humble and understand that others may know more than they do.