

Teamwork instruction

WMSEE 24
Michael Hilton

Context

17-313 Foundations of Software Engineering

~100-120 students a semester

~4 to 6 recitation sections

60% IS, 30% CS, 10% other

Teamwork is a first class learning objective

Teams of size 4-6

<https://cmu-313.github.io/> -> Will share instructor repo if you send me your Github ID



Course outline

14 weeks

First 10 days of the course a Project 1 (individual project to make sure everyone has setup and run the code base, currently NodeBB, OSS Project)

Team Formation

Project 2 -> Team formation, sprints, feature development (5 weeks)

Project 3 -> CI + Deployment (2 weeks)

Project 4 -> Architecture + ML (2 weeks)

Project 5 -> Open Source Excursion (3-4 weeks)

Team Formation

We do not allow students to choose their own teams

We form teams internal to a recitation.

Each recitation has 2 mentor TAs (each TA mentors 2 recitations)

We have 3-4 teams per recitation

We optimize for schedule availability

We also consider students' demographics, program, JS/TS background, likelihood to drop (self reported)

Observation:
Teamwork needs to be
explicitly taught to students

Teamwork Interventions

- Sprint 0 in P2 (first team project)
 - Create Kanban board, issues assigned to team members, marked by TAs
 - Taught branch based development, enforced via rubric points
 - Create and turn in “Teamwork Contract”
 - Extra credit if they do “non-class activity” and send us a picture
 - BETA: Github Actions creates time-bound participation badges
- Weekly teamwork survey: Based on SIGCSE22 paper (Not mine)
 - Use feedback to examine students’ participation
- Recitation 5 (week 5, during P2)
 - Students do an activity where they role-play “bad actors in team dynamics”
- Lecture on giving feedback (Given from business school)
 - SBIA: Situation-Behavior-Impact-Alternative model
 - More info in SIGCSE’24 paper

Teamwork Levers

We encourage students to come to talk with us

Students give each other feedback, but all feedback is graded on participation, not content

If teams report a student isn't participating, then we analyze artifacts more closely

Contributions

We want to ensure that everyone is participating fully in the final project. For this project, we will be assessing participation in a variety of ways, including: artifact evaluation, self & peer evaluation. Credit due for the team components of P5 will be awarded based on evidence of full participation in the team. **Partial participation will receive partial credit.**

If severe teamwork issues arise please contact the course staff.

SIGCSE 24 paper with more details



Improving Software Engineering Teamwork with Structured Feedback

Victor Huang Weiqi

hweiqivi@alumni.cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

Taya Cohen

tcohen@cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

Kori Krueger

kori.krueger@morgan.edu
Morgan State University
Baltimore, Maryland, USA

Michael Hilton

mhilton@cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

ABSTRACT

Teamwork is a key learning outcome for our course, Foundations of Software Engineering. However, conflicts are inevitable in teams, and if students cannot resolve conflicts, this can lead to decreased satisfaction for everyone on the team. In this experience report, we present our approach to help students deal with conflicts that can occur in team projects. Working with faculty from the School of Business, following best practices of organizational behaviour research, we instructed students on how to provide high quality peer feedback, and designed activities where students provided feedback to each other following these principles. After this intervention, we compared the results of our teamwork survey with the results from the previous semester. We saw a meaningful drop in teamwork

unresolved, they can be detrimental to the success of software development projects, and lead to lower developer satisfaction in the team. [3, 11, 17].

In a third year course on software engineering, [Foundations of Software Engineering \(17-313\)](#), students collaborate in teams of 4 to 6 members, working on various projects through the semester. A major recurring challenge is that a portion of student teams struggle to work together. Despite trying various methods to help teams address teamwork issues over the years, the student teams continued to struggle.

"Conflicts with team members" was continuously reported as the largest concern that students had in the class. Moreover, members of such teams tended to report lower satisfaction with the course

