

Building Reflective Student Software Teams

Using the Sprint Report to Turn Setbacks into Growth

Scaffolded Projects for the Social Good (SPSG)

Chad A. Williams, Stan Kurkovsky
Central Connecticut State University

Nathan Sommer
Xavier University

Mikey Goldweber
Denison University



This work is supported in part by NSF awards [2315322](#) and [2315323](#)



Does Any of This Sound Familiar?

- **Common Challenges in Student Software Teams**
 - Teams that struggle to communicate and coordinate
 - Students who over-engineer — or underestimate — scope
 - Difficulty assigning fair individual grades
 - Projects that drift off course mid-semester
 - Students who repeat the same mistakes sprint after sprint



What is the SPSG Framework?



A **multi-institution, NSF-funded framework** that integrates community-based service learning into standard CS courses — using a studio model and agile principles.



Designed to be:

Low adoption threshold

Any instructor

Any institution



At a glance

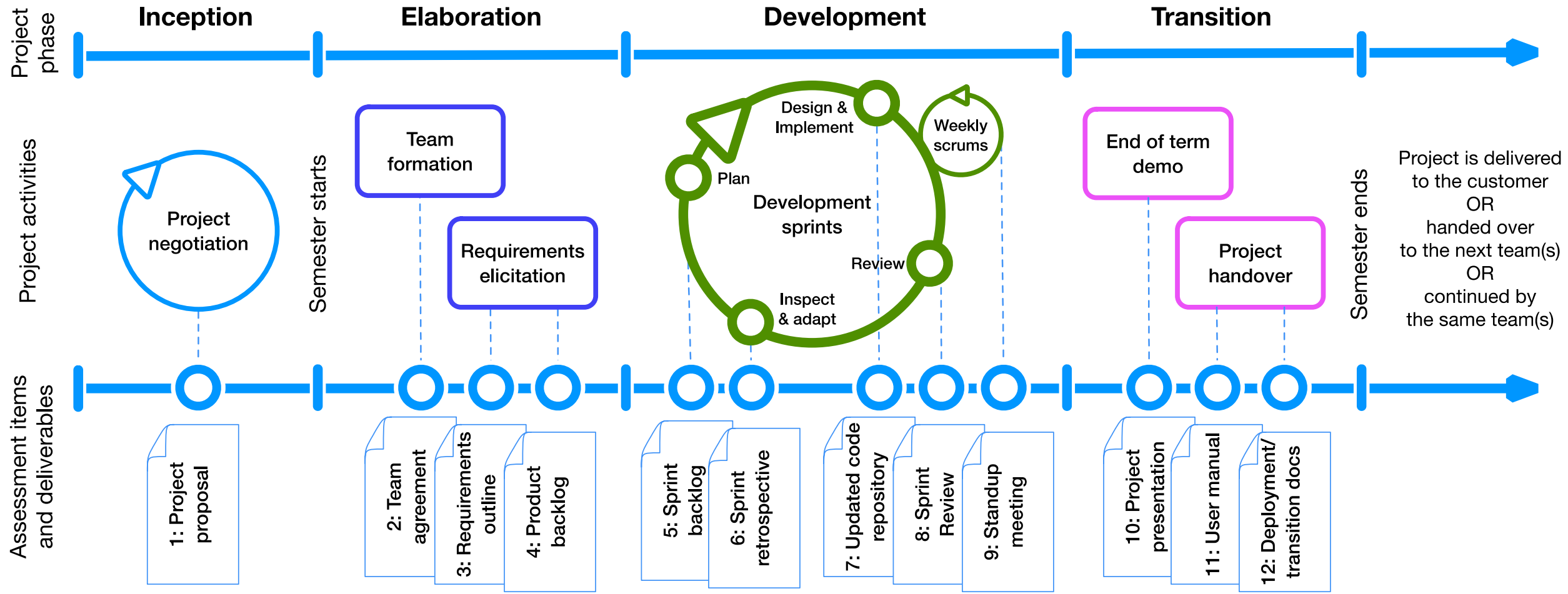
650+ students engaged to date

10+ years of development

75+ real-world community partners

The SPSG Roadmap

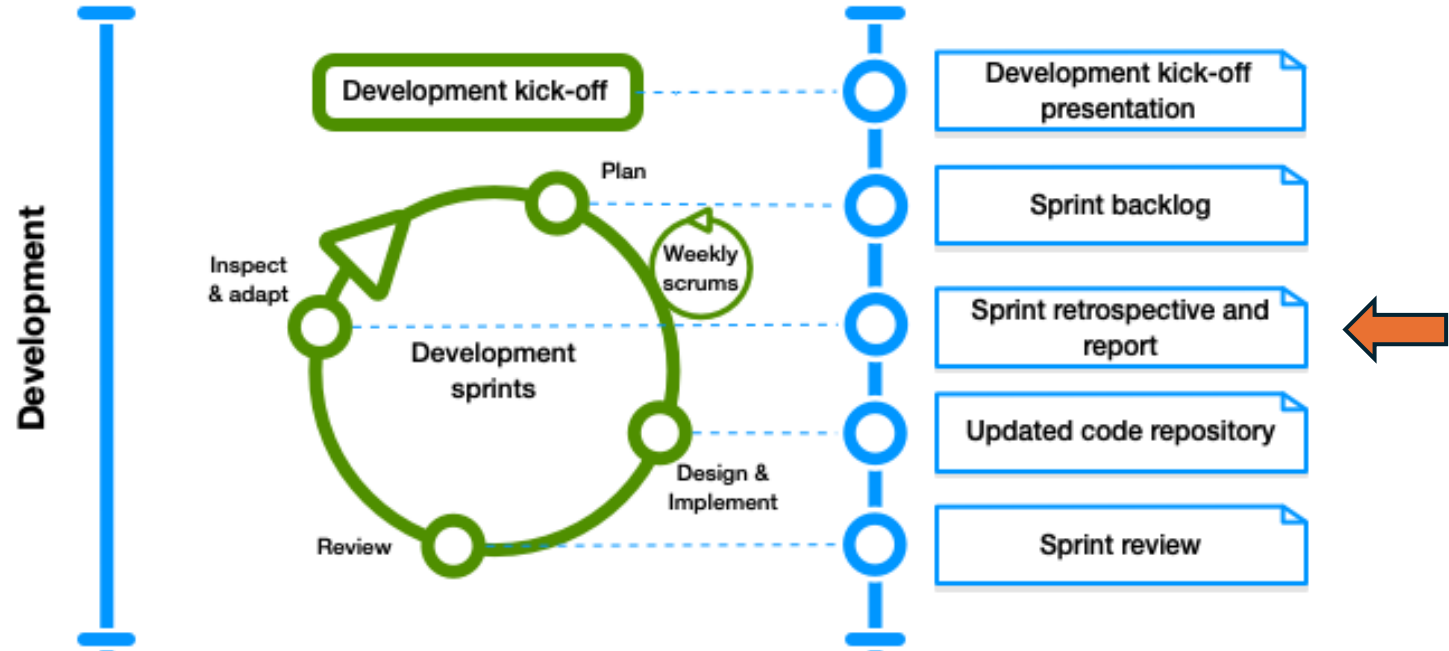
Each phase has scaffolded deliverables — structured support without removing student agency.



Zooming In: The Sprint Report

By this point, teams have already established their norms, gathered requirements, and built their backlog. The Sprint Report is what keeps everything honest from here.

- **The Green Loop — Repeated Every Two Weeks**
- **Deliverables inside the loop:**
- **5:** Sprint Backlog
- **→ 6: Sprint Retrospective ← (our focus today)**
- **7:** Updated Code Repository
- **8:** Sprint Review (demo with project partner)
- **9:** Standup Meeting



What the Sprint Report Captures

Accountability	Reflection & Adaptation
Planned vs. completed story points	What worked? What didn't?
Individual contributions — visible to the whole class	Lessons learned → concrete changes committed to
In-class retrospective presentation	Updated backlog for the next sprint
Demo of working functionality	Anticipated challenges for next sprint

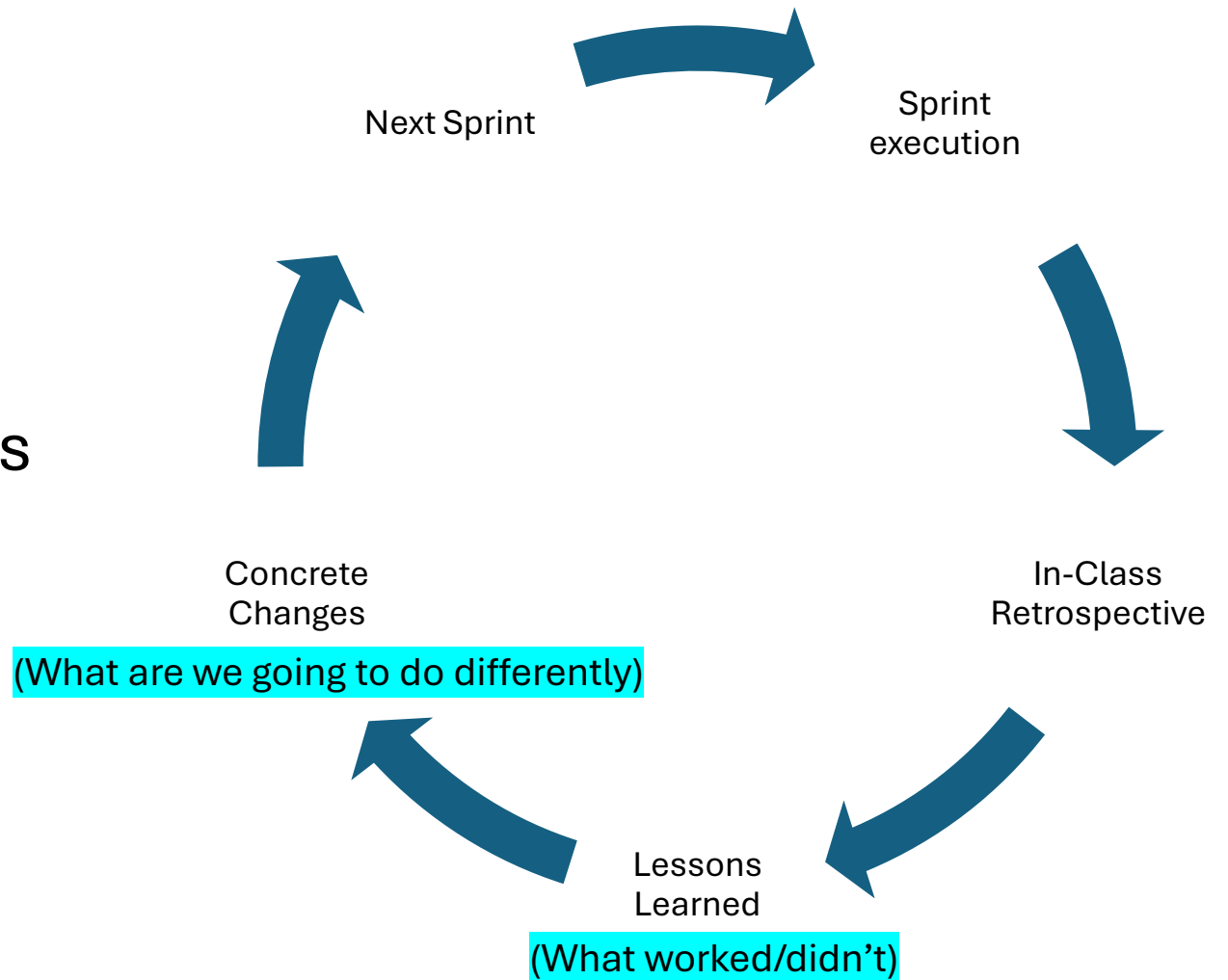
The most important question in the template

"What changes will you be making based on the lessons learned?"

The Reflection-to-Action Loop

Key Insight

- Students don't just report problems – **they change behavior** next sprint
- Transforms technical failures into **professional growth**



Ex. Team Mad Hatterz: Growth Across Six Sprints

	Phase	Challenge	Lesson Learned
Sprint 1	Early: Technical Troubleshooting	Struggled for days with a new XML parser that failed to work as expected	"Newer isn't always better — go with trusted technology. Do thorough research before adopting new libraries."
Sprints 2–3	Middle: External Blockers	Meta API permission hurdles; no Apple Developer ID stalled multiple sprints	"Pivot quickly. Plan more conservatively to allow time to catch up from unforeseen blockers."
Sprints 5–6	Late: Risk Management & Professional Standards	Last-minute branch merges created visual artifacts in the demo; a destructive database query nearly derailed final delivery	"Merge early. Never make destructive database queries close to a deadline. Implement backups and rollback measures."

These Lessons Are Universal

The Same Realizations. Different Projects. Every Semester.

- **The Over-Engineering Trap** – Teams independently concluded they chose overly complex paths - learn need for balance
- **Time Management Realism** - All reached the same conclusion: complex, high-priority tasks must be tackled early
- **Communication Breakdown** - Identified miscommunication as their biggest hindrance — each independently concluding that consistent, open communication is "key to a successful project."

The cross-team convergence is actually pedagogically meaningful: students aren't being told these lessons are important, they're discovering them independently, which is how lasting professional knowledge is built

Takeaways & Next Steps

What to Take Away

- **1.** The Sprint Report is a repeatable, low-overhead mechanism that transforms project setbacks into structured, documented learning opportunities — with evidence of behavior change across sprints.
- **2.** It is one element of a comprehensive framework built specifically to address the challenges we've been discussing today — team accountability, scoping, communication, and fair evaluation.
- **3.** Everything is freely available: templates, instructor guides, and student examples

<https://spsg-hub.github.io/>

