Section 09 – Discussion Topics

How is Software Engineering different from other classes and disciplines you have been exposed to?

- Holistic nature of the subject
- High level issues – beyond the technical ones, but at least as important
- Not many clear answers; mostly good practices

Students suggested the following differences:

- Increased student responsibilities for the project (from A to Z)
- Emphasis on the process (not the product)
- No theory / science, “squishy” material
- Working in large teams
- Scale of activities
- Potential applicability of projects
- No fixed base on top of which to build (i.e., the foundation keeps changing underneath you)

What aspects has this class not exposed you to that occur in “real world” software development environments?

- Changing requirements
- Real outside customers
- Working full-time on a major software project
- People leaving and joining the team partway into a project
- Monetary compensation

Students suggested the following aspects:

- Combination of products from other disciplines in order to reach the project goal (e.g., developing both hardware and software)
- Organization hierarchy and its consequences (hiring, firing, promotions, demotions, etc.)
- Different software markets and their broad requirements
- Working not from scratch

Do the above distinctions make it easier or harder to succeed in the industry?

- Solid, tried-out processes are all the more important when more unknowns enter the picture. Otherwise, there is a real chance of chaos.

Key lessons from the “Mastery” book:

- The importance of life-long learning
  - If you exchange $1 with someone, each of you still has $1. If you exchange an idea, each of you now has 2 ideas.
  - “You can win, you can lose, or you can learn.” (V. Satir)
  - “Since you’ve paid the tuition, you might as well take the lesson.” (J. Weinberg)
- It’s okay to be a “fool” (i.e., to admit that you are not an expert)
  - Opens up your mind to new directions and possibilities
- The plateau metaphor
  - Spikes are rare, so learn to love the plateau
  - Faux promises: “instant success”, “fast relief”, “endless climax”, “immediate gratification”
- The goal is the experience and continual practice, not reaching perfection or some final stage.
  - “There’s no way to happiness; happiness is the way.”
  - “— Excuse me, how do I get to Carnegie Hall?” “— Practice!”
  - Mastery is a journey, not a state.
- The homeostasis phenomenon (biological and social)
  - It’s safer to stay where you are than to change; changing is risky.
- How do the above tie to software engineering?