

Lecture 1: August 27

Introduction to Senior Design (CS 4243W)

Agenda / Topics

Course Review

Website Review / Setting up Slack

Project Management Methodologies

For Next Week

Forming Teams

About Me



About Me



Why am I here?

Goals for Senior Design

1. Build a cool project that represents the culmination of 4 years of CS

Goals for Senior Design

1. Build a cool project that represents the culmination of 4 years of CS
2. Prepare you for life after GW

Goals for Senior Design

1. Build a cool project that represents the culmination of 4 years of CS
2. Prepare you for life after GW
 - Presentation skills
 - Writing skills
 - Communication skills
 - Industry engineering practices

Goals for Senior Design

1. Build a cool project that represents the culmination of 4 years of CS
2. Prepare you for life after GW
 - Presentation skills
 - Writing skills
 - Communication skills
 - Industry engineering practices
 - Figure out what's next – academia, industry, anything other than CS

Why am I here?

Why am I here?

1. I went to GW and took your Senior Design Course

Why am I here?

1. I went to GW and took your Senior Design Course
2. I have industry experience

Why am I here?

1. I went to GW and took your Senior Design Course
2. I have industry experience
3. It took me a while to figure out what I wanted to do

Instructors

Professor Qu

Professor Kim

Josh Shapiro

Industry Mentors

Aisha Mohammed	'23	MITRE
Austin Blackman	'15	Oracle
Billy Miller	'21	MITRE
Brandon Harvey	'23	Marine Acoustics
Connor Burnett	'23	Alarm.com
Dennis Afanasev	'20	Oracle
Dietrich Reidenbaugh	'19	Amazon
Elena Korkes	'23	GWUSEC
Eli Mattson	'14	Indigov
Gregor Peach	'20	Amazon
Jonathan Lee	'23	CMU Robotics
Leah Staniorski	'15	Stripe

Max Englander	'23	Capital One
Pat Cody	'20	Meta
Phil Lopreiato	'17	Facebook
Ravi Goti	'13	eBay
Rian Shambaugh	'15	FINRA
Victoria Hennemann	'23	GWU Cybersecurity

Course Review

- Tuesday class
 - Regularly scheduled lecture will **not** meet most weeks
 - Each team will have weekly online meetings w/ faculty advisor
 - Progress check in, feedback on technical & algorithmic complexity
- Wednesday lab
 - 6:10-7:30(ish): lecture*
 - 7:30(ish)-8:40: work with team on project & meet with mentor
- Mentor meetings (weekly)
 - Weekly “standup” & monthly “sprint planning”
 - Mentor acts as scrum owner
 - Mentor provides guidance on technologies & best-practices
 - Schedule time w/ your mentor (doesn't have to be Wednesday night)

Wednesday Lab - Course Topics

1. Lectures (project management & technical skills)
2. Industry presentations / alumni panels
3. Interview prep
4. Team presentations / demos
5. Instructor feedback
6. Project development

Fall Semester Grade Breakdown & Expectations

- Presentations **35%**
 - Presentation 1: Elevator Pitch **5%**
 - Presentation 2: Technical Design **10%**
 - Presentation 3: Alpha Presentation **20%**
- Writing assignments **20%**
- Sprint Progress **20%**
 - Weekly status updates
 - Sprint boards
 - Code commits
 - End of sprint demos
- Alpha Project Review **20%**
- Participation **5%**
 - Attendance
 - Team surveys / homeworks

Time Commitment (per semester):

- **55 hrs** in class/lab
- **150 hrs** out of class working on project

Agenda / Topics

Course Review

Website Review / Setting up Slack

Project Management Methodologies

For Next Week

Forming Teams

Course Website

1. [Join Slack](#)
2. Schedule & Due Dates
3. Grade Breakdown
4. Project Requirements

<https://go.gwu.edu/cscapstone>

Agenda / Topics

Course Review

Website Review / Setting up Slack

Project Management Methodologies

For Next Week

Forming Teams

Project Management

Project management is the use of specific knowledge, skills, tools and techniques to deliver something of value to people.

Generally,

- No 2 projects are identical
- No 2 teams are identical

Therefore, methodologies must work to provide the best way to deliver on that value with the team at hand

Top Project Management Methodologies

1. Waterfall
2. Agile
3. Scrum
4. Kanban
5. Scrumban
6. eXtreme programming
7. Adaptive Project Framework
8. Lean

Waterfall

Traditional approach where tasks and phases are completed in a linear, sequential manner, and each stage of the project must be completed before the next begins

Works well if...

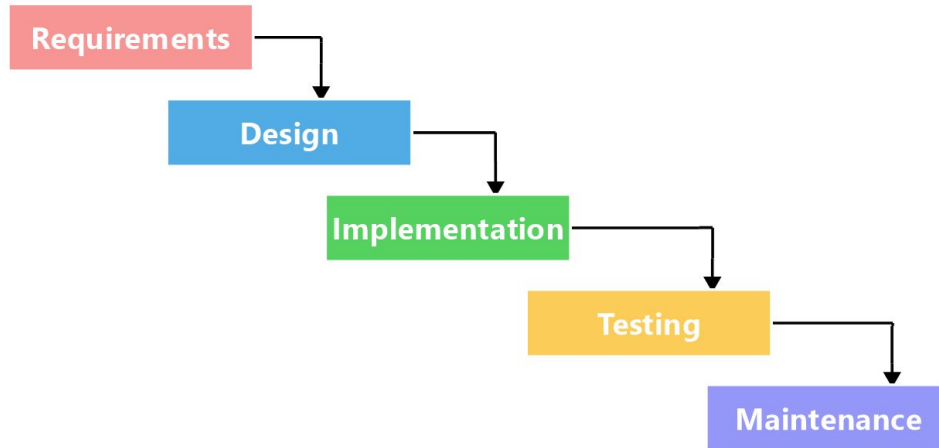
- End goal is clear
- Stakeholders know exact needs
- Project is consistent
- Working in well-regulated industry

Doesn't work well if...

- Project is liable to change
- You don't have full requirements
- You need continuous feedback

Waterfall Methods

- Kick off meeting to collect requirements
- No more customer touch points after this meeting
- Design, Implement, Verify, and Maintain to completion



Agile

The agile project management methodologies usually involve short phases of work with frequent testing, reassessment, and adaptation throughout

Works well if...

- Your project is liable to change.
- You're not sure at the outset what the solution will look like.
- You need to work quickly, and it's more important that you see speedy progress than perfect results.
- Your stakeholders or client needs (or wants) to be involved at every stage.

Doesn't work well if...

- You need a predictable deliverable, and you need to be crystal clear about what that looks like from the outset.
- Your project can't afford to change during its course.

Agile

History Blurb: ["The Agile Manifesto"](#)

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

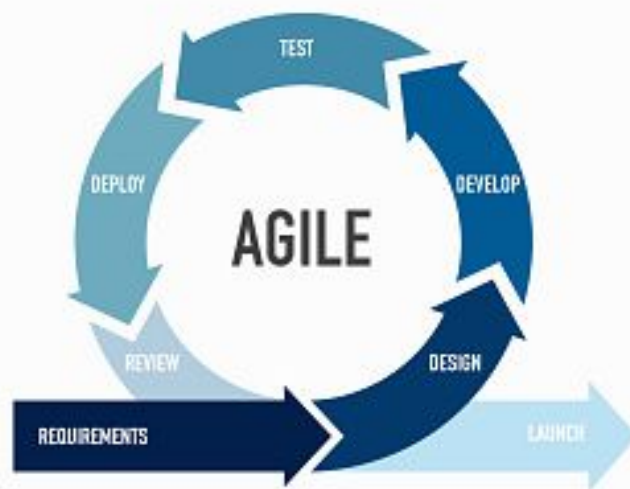
Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

AGILE vs WATERFALL



Agile Concepts

Sprint:

Short, repetitive period of time in which the team commits to a certain amount of work (in some cases points) to be completed in that time.

Backlog:

List of all tasks that are required for the project completion (project backlog) or other goals (eg. Tech Debt Backlog)

Sprint Board:

Place to track the tasks being worked on for the sprint

Agile Processes and Rituals

One 2-week Sprint Includes...

1 Sprint Planning Meeting: Bring backlog items into sprint board

Daily Standups: Checkin everyday to unblock each other

1 Backlog Refinement: Clean up backlog to prep for next sprint(s)

1 Sprint Review “Demo”: Demo work done in the sprint

1 Sprint Retrospective: Honest conversation about what went right, wrong, and action items to improve

***this is not a perfect recipe, many teams will adjust to what works best*

A Typical Monthly Sprint Schedule

Week 1

Day 1

Sprint Planning Meeting

Days 2-7

Standup Meeting

Work on project: update cards, use slack to unblock and questions

Week 2

Days 1-7

Standup Meeting

Work on project: update cards, use slack to unblock and questions, etc

Week 3

Days 1-7

Any day this week

Standup Meeting

Work on project: update cards, use slack to unblock and questions, etc

Backlog Refinement

Week 4

Days 1-7

Day 7

Standup Meeting

Work on project: update cards, use slack to unblock and questions, etc

Demo, Retrospective

Using Agile for our Projects

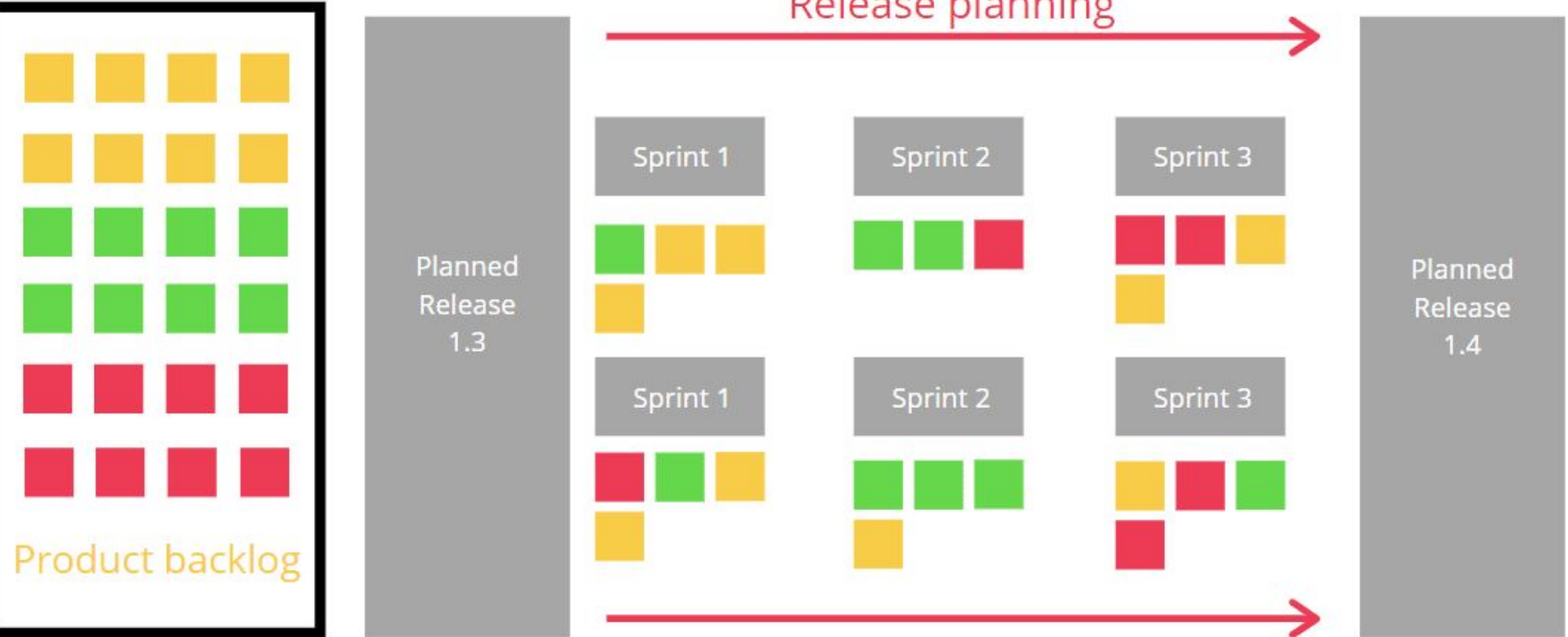
Why?

- Not all requirements are known
- Stakeholders are flexible on product-specific requirements
- Project duration requires iterative sprints of work
- Enabling a wide variety of projects against the same course goals

How?

- Github Projects as the source of truth
- Mentors as Scrum Owners and “Technical Consultants”

Agile Mapping



Scrum, Kanban, and Scrumban

Scrum

Work is split into short cycles known as “sprints”, which usually last about 1-2 weeks.

Focus: Deliver potentially shippable increments of progress at the end of each sprint

Kanban

Tasks are visually represented as they progress through columns on a kanban board. Work is pulled continuously from the backlog.

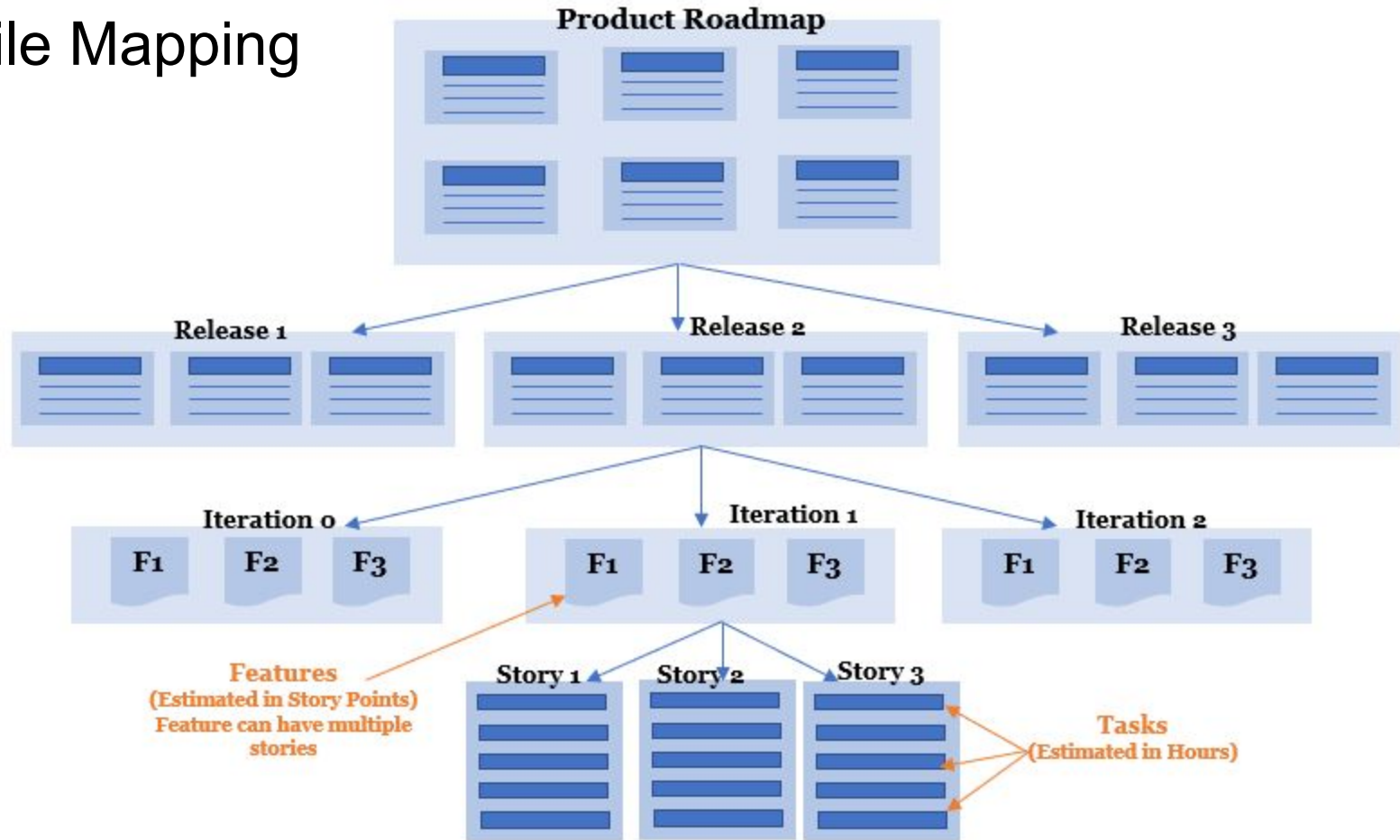
Focus: Improve efficiency and throughput by limiting WIP.

Scrumban

Work is split into short cycles known as “sprints”, which usually last about 1-2 weeks. Work is pulled continuously from the backlog.

Focus: Adaptable to changing priorities & continuous improvement.

Agile Mapping



Project Manager

Who?

- A Project Manager is responsible for the planning, procurement, execution and completion of a project
- In charge of the entire project and handles everything involved, such as the project scope, managing the project team, as well as the resources assigned to the project.

Roles and Responsibilities?

- Responsible for Success or Failure of project
- Github Projects as the source of truth
- Mentors as Scrum Owners and “Technical Consultants”

Scrum Owner

Who?

- Lead for the duration of the sprint, after which they review their performance in a “sprint retrospective” and make any necessary changes before starting the next sprint.

Roles and Responsibilities?

- Github Projects as the source of truth
- Mentors as Scrum Owners and “Technical Consultants”

“Agile-ish. We have daily stand ups and weekly planning meetings and we iterate over requirements, but we don’t work in sprints.”

Rian, FINRA

“We use agile with two week sprints, and quarterly planning based around OKR’s”

Lucas, Square

“We do planning every 2 weeks but don’t really follow any methodology besides that”

Aaron, Meta

Quick Poll:

“It’s complicated (ranges from sprintish planning, to 6 month roadmaps, to nothing at all)”

Phil, Meta

@Mentors, what method does your team use?

“Strictly Agile, more specifically scrum flavored: biweekly sprint planning coupled with demo-able "stuff" we completed the past sprint. "If it goes into the sprint it should be completed, otherwise, you've overestimated”

John, Toyota

“For my team specifically, we don’t really have any structure beyond a 6 month roadmap, it’s up to the individual and how they want to break up the work.”

Pat, Meta

Project Management: Basic Tools

Project Management Tools

For tracking tasks...

- Trello, ASANA, GitHub (very small teams)
- Jira (established, large teams)

For collaborating on documents...

- Confluence
- Notion
- Coda
- Google Drive Suite: Docs, Sheets, Slides, etc.

“Jira and confluence”
Rian, FINRA

“We use google docs/sheets”
Aaron, Meta

“Jira for story management and
coda for docs/wikis/planning”
Lucas, Square

Quick Poll:

“Custom tasks tool (works like
jira, but better) and gdocs for
most of it”
Phil, Meta

**@Mentors, what project management tool does
your team use?**

“Jira for all project tracking and
logistics. Confluence for any docs
(although in research docs are a
rare luxury)”
John, Toyota

“Custom task tools plus
GDocs (RIP quip)”
Pat, Meta

Major Takeaways

- Two most common methods: Waterfall and Agile
- Software mainly uses Agile
- Methodologies are not mutually exclusive (eg. you can be “agile” in a waterfall process)
- A Project Manager, Scrum Owner, or Product Manager owns this process
- Trello, Jira, Confluence, and other “Atlassian” projects are industry standard for tooling in this space

How this applies to Senior Design

- We will be using a “flavor” of Agile and Scrum for our projects
- Github Projects is our source of truth – mentors & faculty will use this to measure team progress. You’ll include weekly status updates here.
- Sprints will be 1 month long
- Use slack for meetings with mentors & technical discussion

Agenda / Topics

Course Review

Website Review / Setting up Slack

Project Management Methodologies

For Next Week

Forming Teams

For Next Week

- Confirm [access to slack](#) & upload a photo
- Complete [student info form](#) ASAP
- Complete [team submission form](#) (one per team!) before Tuesday's class (Sept. 2)

Agenda / Topics

Course Review

Website Review / Setting up Slack

Project Management Methodologies

For Next Week

Forming Teams

Form a team!

- 4 students
- Similar ideas for projects
- Variety of specializations/experience
- Align project ideas with past coursework
- > 200 hrs per semester working on this project, choose a project you care about!

Submit [team formation survey](#) before next Tuesday!