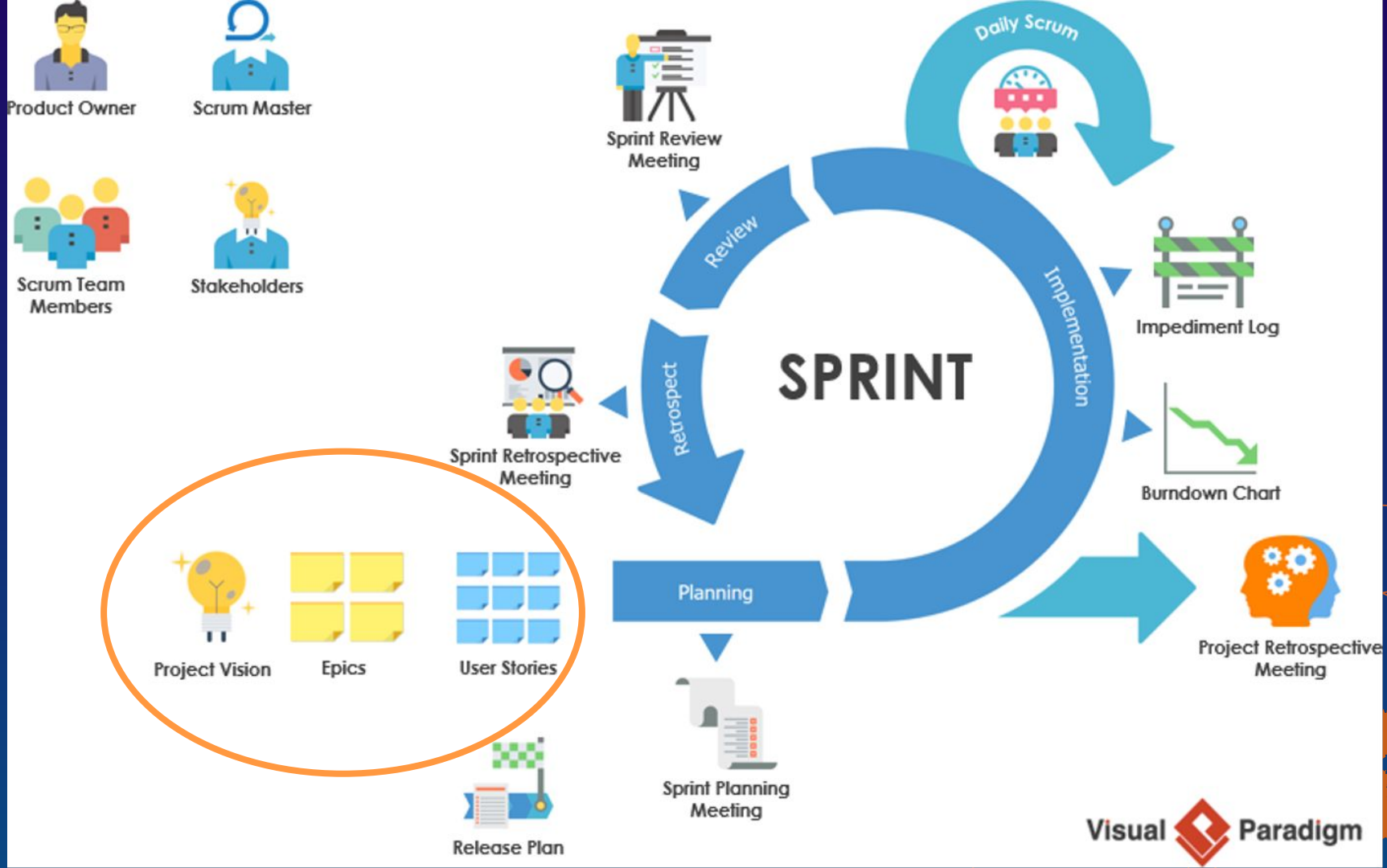
W05-2: Teaming & Sizing Sprint 0 Kickoff

Dr. Alex Adkins

CPSC 3720: Software Engineering

Today’s Objectives

- Estimating User Stories
- Importance of Teams in Software Engineering
- Project Kickoff!


Product Owner
Scrum Master
Scrum Team Members
Stakeholders
Sprint Review Meeting
Daily Scrum
Review
Implementation
Retrospect
SPRINT
Impediment Log
Burndown Chart
Sprint Retrospective Meeting
Project Vision
Epics
User Stories
Planning
Project Retrospective Meeting
Release Plan
Sprint Planning Meeting
Visual Paradigm

Recap: User Stories

What are User Stories?

User Stories:

- short simple descriptions of a features
- told from the perspective of the person who desires the new capability
  - usually a user or customer of the system

User Story Template:

As a <type of user/user role>, I want <some goal> so that <some reason>

WHO **WHAT** **WHY**

User Stories: General Guidelines

INVEST

- Independent
- Negotiable
- Valuable
- Estimatable
- Small
- Testable

- S.A.M. (helps with testable)
  - Specific
  - Attainable
  - Measurable

Story Hierarchy

- Product: the largest chunk of value
- Epic: a medium chunk of value
  - larger than a story but smaller than the whole product
- User Story: smallest piece of value
  - can be implemented in at most a **week or two**
  - **broken down into tasks for each sprint**

Stories should answer: Who, What, Why?

- Who - the the user story is for (As a User)
- What - the Functionality that the user story implements (I want)
- Why - the reason the user needs the user story (so that)

User Stories: User Roles

- **User Role:** a collection of defining attributes that characterize a population of users and their intended interactions with the system

Source: *Software for Use* by Constantine and Lockwood (1999).

User Stories: Acceptance Criteria

- **Acceptance Criteria**: the conditions that must be met for the work to be complete

Acceptance Criteria Example

User Story

As a customer, I want to transfer money between my accounts so that I can manage my finances easily.

Acceptance Criteria

- **When the customer selects the Transfer option and chooses the source and destination accounts, the system should display the available balance in the source account.**
- **The transfer amount must be less than or equal to the available balance.**
- **When the customer confirms the transfer, the system should immediately deduct the amount from the source account and add it to the destination account. It should then display a confirmation message on the screen.**
- **The transfer must be logged with a timestamp and a unique transaction ID.**

Acceptance Criteria Example

User Story

As a team member, I want to log my work hours on tasks so that I can track and bill my time accurately.

Acceptance Criteria

- **The system should allow the team member to log hours worked on each task, with an option to specify the date and time period.**
- **The logged hours must be visible to both the team member and the project manager, with the ability to edit or delete entries as needed.**
- **The system should automatically calculate the total hours worked per task and display this information in the task details.**
- **The user should be able to export logged hours to a timesheet for billing purposes.**

Acceptance Criteria Example

User Story

As a product owner, I want to prioritize the product backlog so that the most valuable features are developed first.

Acceptance Criteria

- **The system should allow the product owner to assign a priority level to each user story in the product backlog.**
- **When the priority is updated, the system should automatically reorder the backlog to reflect the new priority.**
- **The product owner should be able to filter stories by priority, status, and assigned team members.**
- **The system should also allow the product owner to add notes or comments to each story, explaining the reasoning behind the priority level.**

TigerChow: Creating Epics & Stories

001 Winner

37 respondents

51 %

002 Winner

36 respondents

49 %

TigerChow: Creating Epics & Stories

We've been hired to build TigerChow.com!

- TigerChow is new Clemson online food delivery service.
- TigerChow should be available via the web and mobile devices.

GAME!

- **In your teams, use the roles on the next slide to brainstorm epics/stories.**
- **Document one epic/story per role with acceptance criteria.**
- **Write them down.**

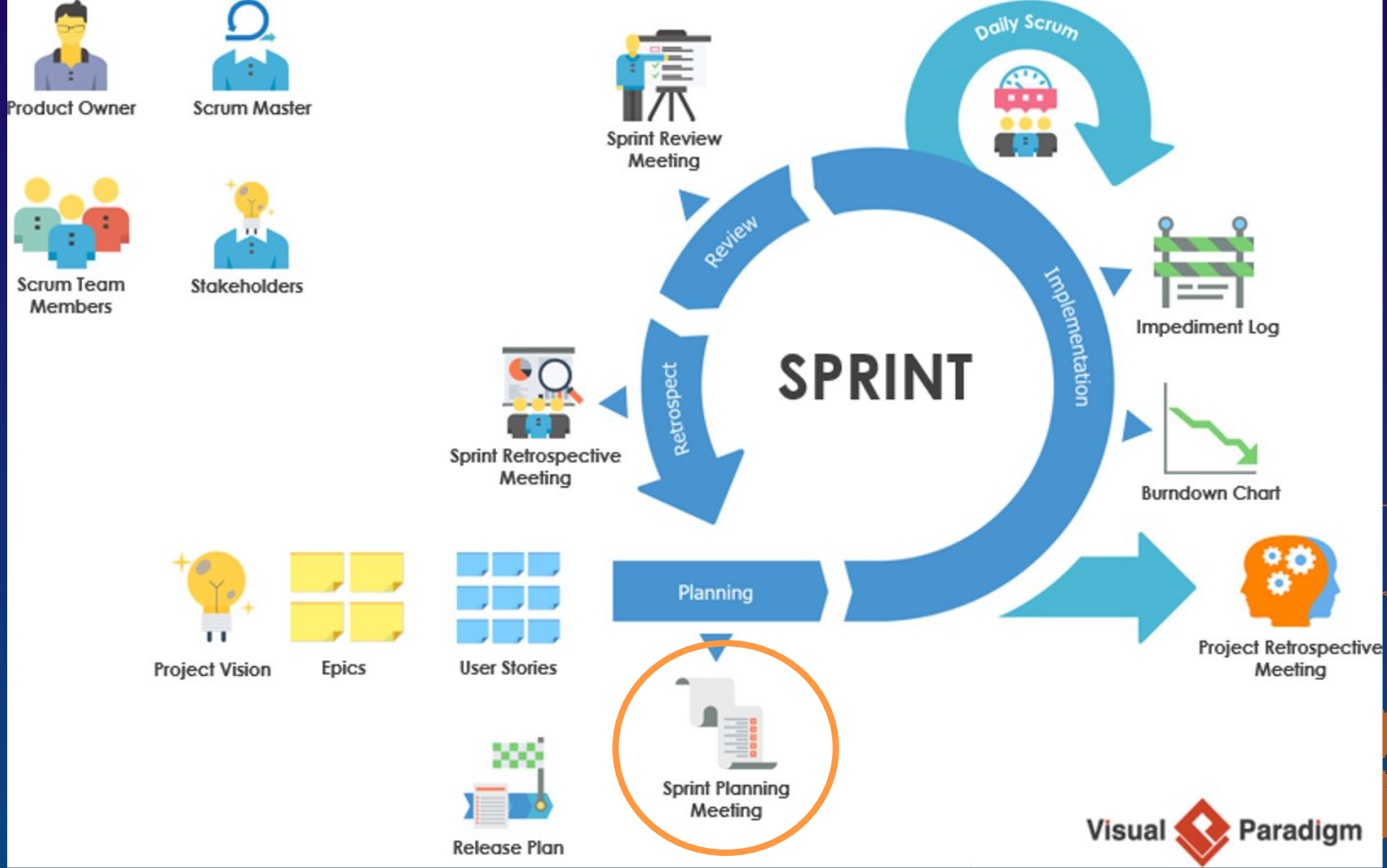
TigerChow Roles

GAME!

- **In your teams, use the roles to brainstorm epics/stories.**
- **Document one epic/story per role with acceptance criteria.**
- **Write them down!**

- Customer: orders food
- Restaurant Staff Member: sets up menu, prepares orders
- Application/System Admin: manages and monitors the TigerChow application
- Deliverers: pick up food from restaurant and delivers to customers
- Customer Service Representatives (CSR): addresses questions and complaints from the customers
- Business User: tracks sales, customer satisfactions, restaurant performance

Planning & Estimation


Product Owner
Scrum Master
Scrum Team Members
Stakeholders
Sprint Review Meeting
Daily Scrum
Review
Implementation
Retrospect
SPRINT
Impediment Log
Sprint Retrospective Meeting
Burndown Chart
Project Vision
Epics
User Stories
Planning
Project Retrospective Meeting
Sprint Planning Meeting
Release Plan
Visual Paradigm

So, we have some Epics/Stories…

- What will it take to deliver the stories?
- How can we *estimate* the amount of work?

checkin.cs.clemson.edu:

Discussion

In your teams…
(3 minutes)

How does Agile help with planning?

How does Agile help with planning?

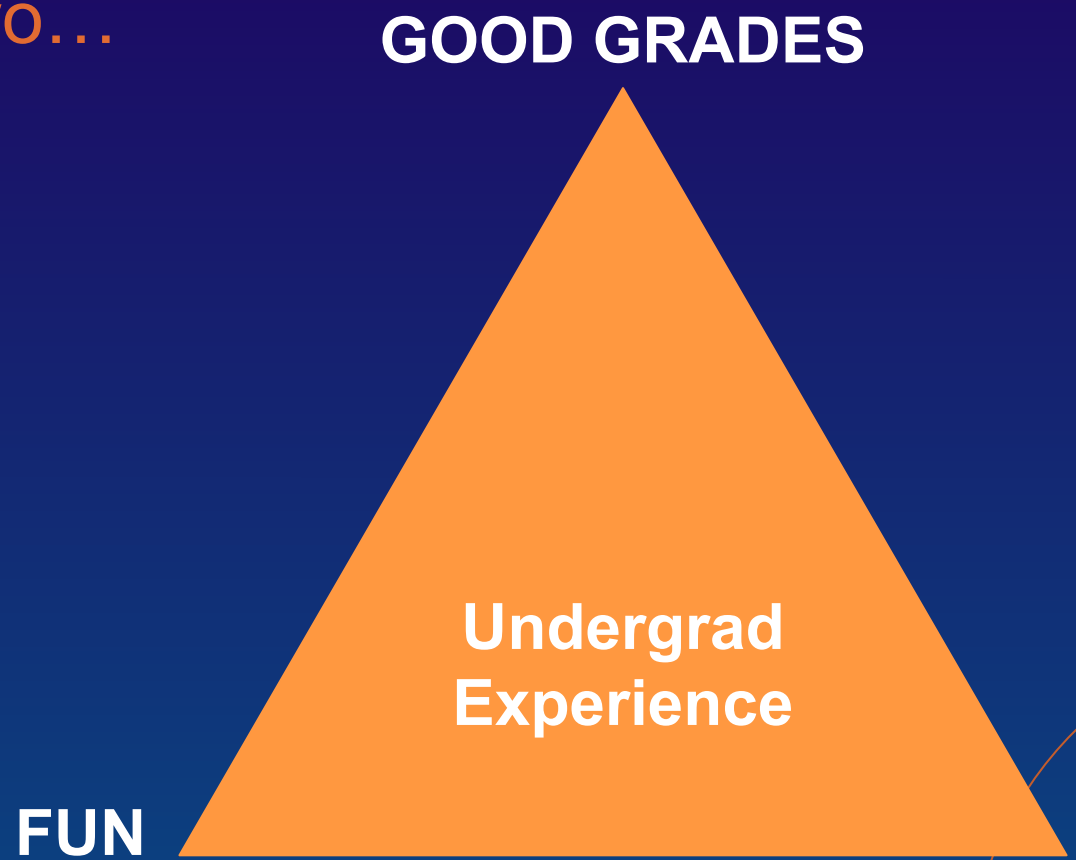
- Agile embraces the uncertainty in software development

- Agile focuses on creating plans that:
  - Are highly honest and visible
  - Focus on customer value in the shortest amount of time
  - Allow you to change course
  - Enable frequent progress reviews with each sprint

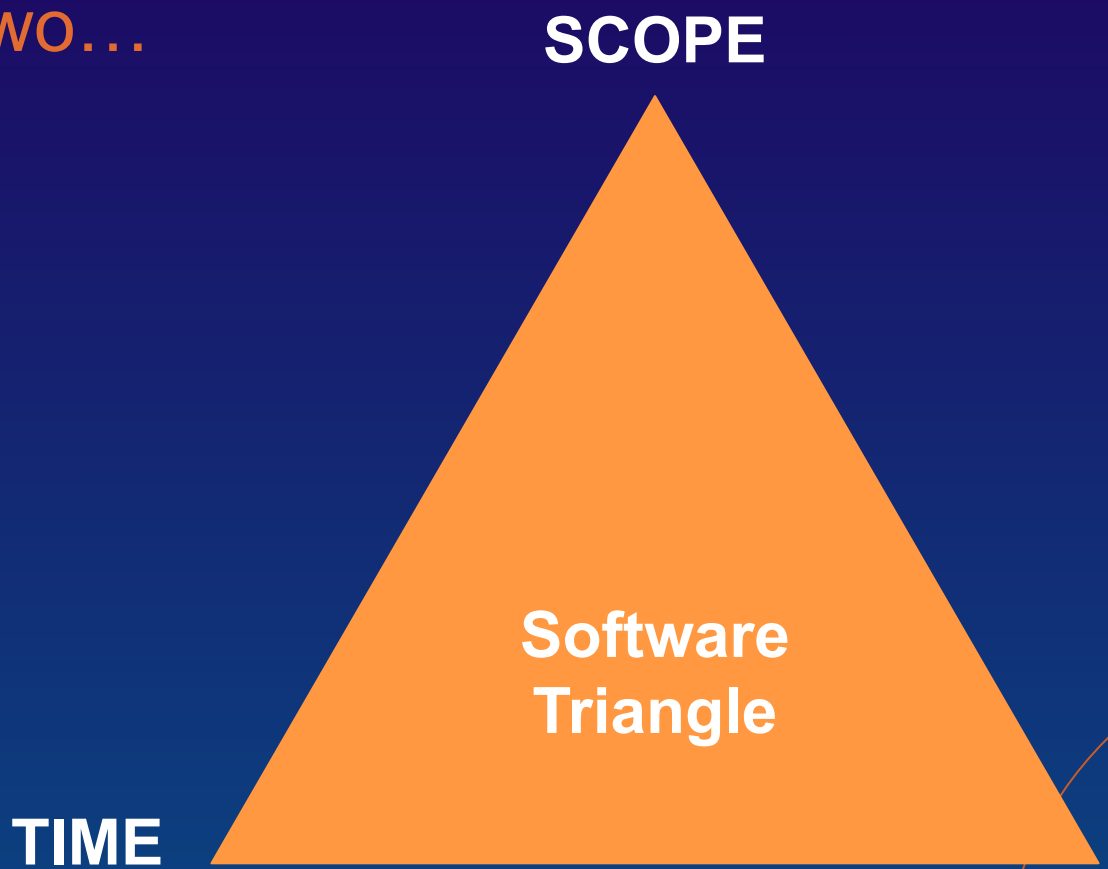
Agile vs. Traditional Estimation

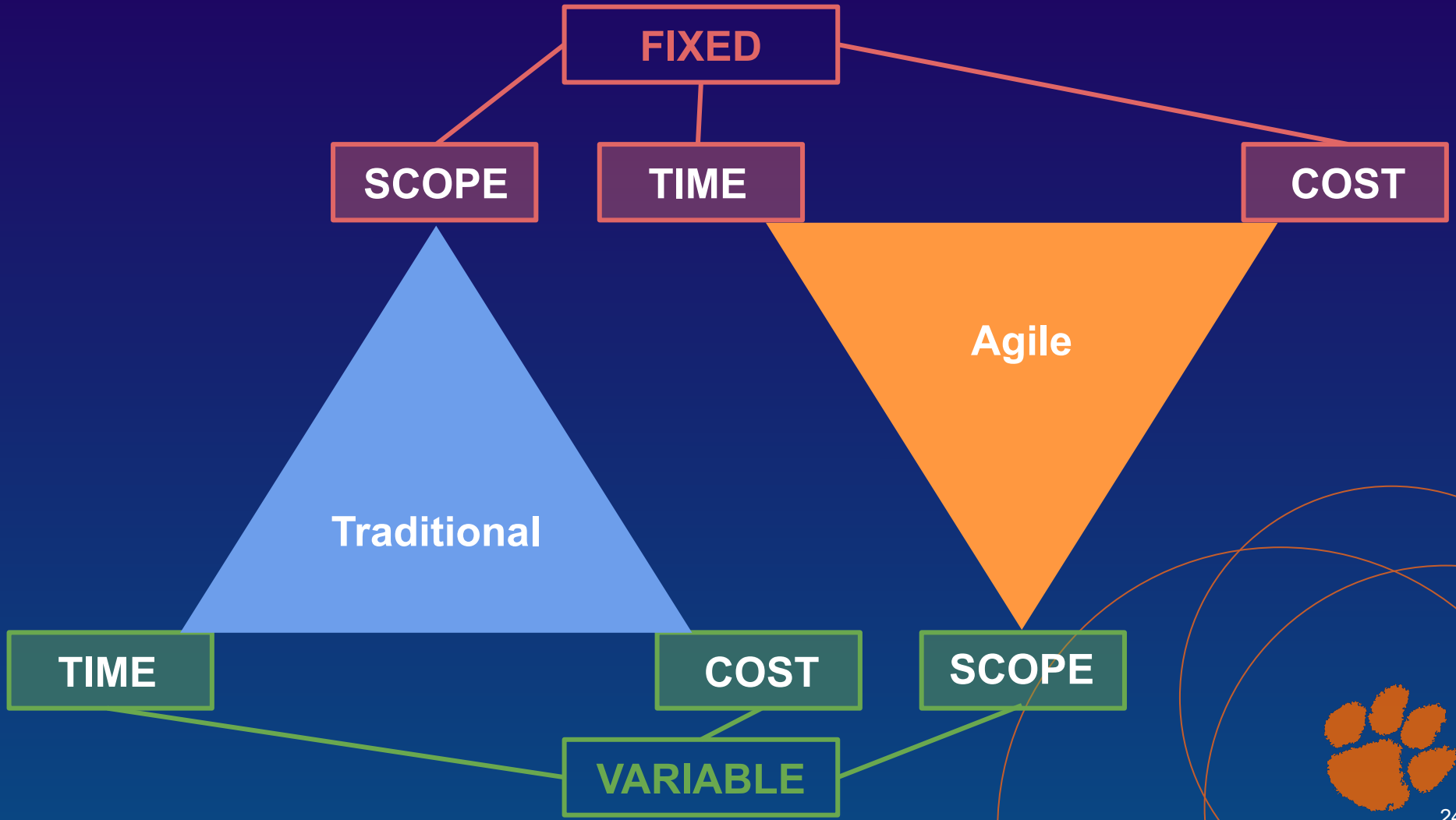
Traditional Estimation	Agile Estimation
Estimation of work to create a timeline	Estimate the story for its value and complexity
Estimate focus is to determine when we will get the entire project/product done	Estimate focus is on how much business value we can deliver in each sprint
Estimate each area separately (dev, testing, documentation, etc.)	Estimate all work as required to deliver the story
Estimate using hours or days	Estimate using relative sizing

Pick Two…



Pick Two…




FIXED
SCOPE
TIME
COST
Traditional
Agile
TIME
COST
SCOPE
VARIABLE

Much of Software Development is Sequential

“The bearing of a child takes nine months, no matter how many women are assigned.”

OR

“9 women in a month cannot make a baby”

Fred Brooks Planning Rule of Thumb

1/3 planning

1/6 coding

1/4 component test and early system test

1/4 system test, all components in hand

Agile Estimating

- Agile understands that nobody can predict the future!

- The team does the estimating: they know best!
- Agile estimate by size and not by time
  - Sizing is relative and not absolute
  - Sizes can be measured in various ways, e.g “Story” points
    - Points are just a number
- Estimate using:
  - Past knowledge
  - Expert advice

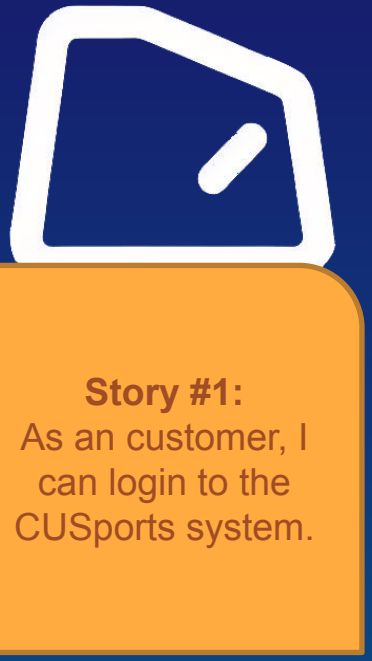
- Remember – it is just an estimate!

AGILE Estimating - The Estimate Includes Everything

UX

Design

Testing

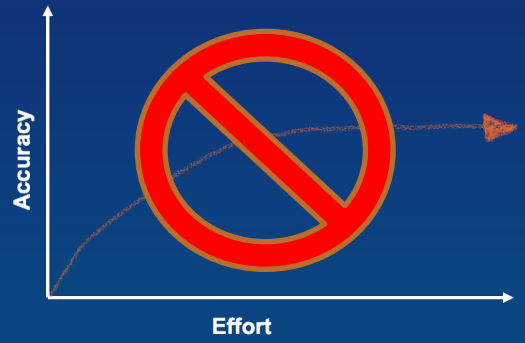


Coding

Documentation

Plans need Estimates

- How?
  - Keep it Simple
  - Use Relative Sizing
- Once you have estimates:
  - Create a high-level plan and budget
  - Ongoing refinement (reminds us we were guessing)
  - Don’t get stuck in Analysis Paralysis!
    - It’s better to have an inaccurate estimate than no estimate



Use Relative Sizing

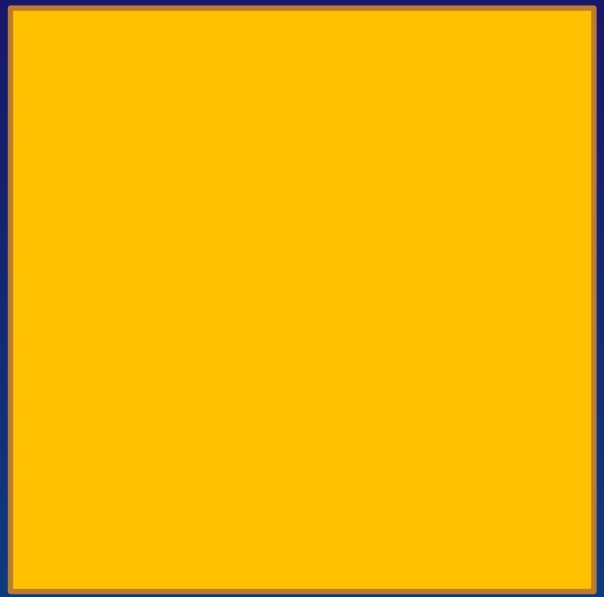
Small Story

Medium Story

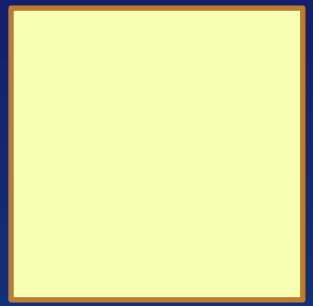
Large Story

Epic or Product

Estimating Absolutely

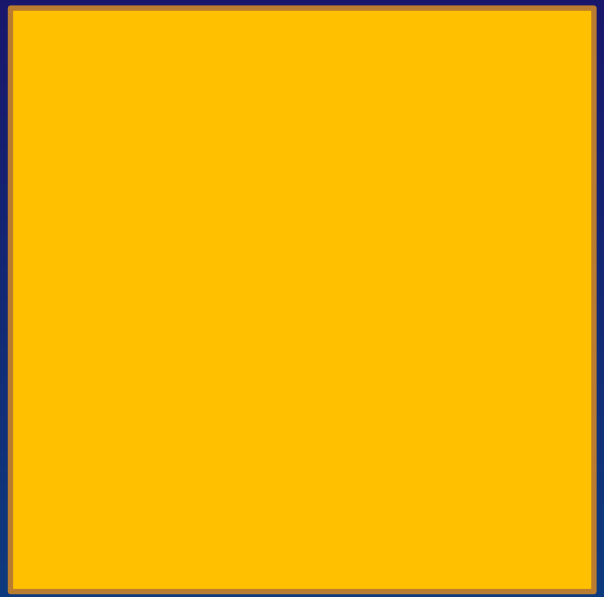


X in^2

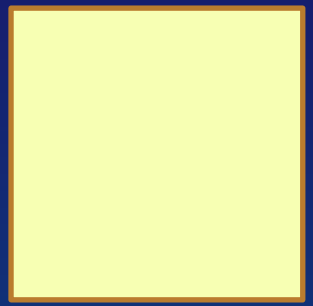


Y in^2

Estimating Relatively

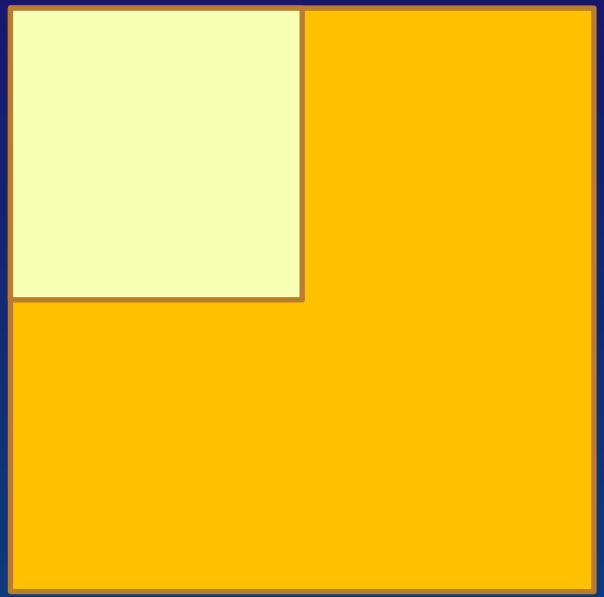


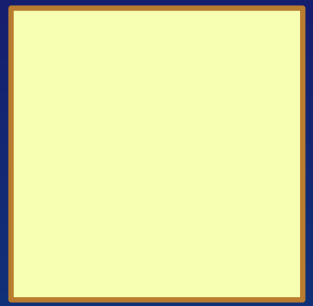
4 pts



1 pt

Which is easier?





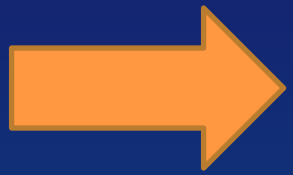
4x

Relative Sizing is a Cornerstone of Agile Planning

Think about “bigness” of a story and not “time”

Units of Measure Don’t Matter!

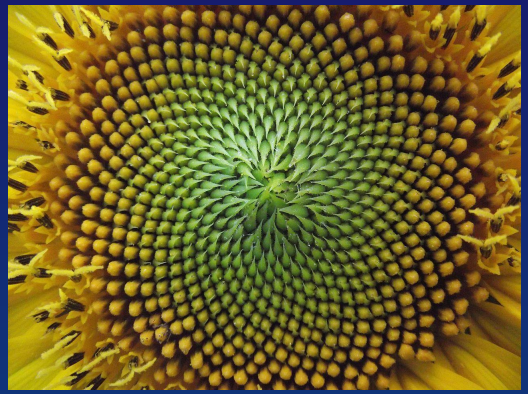
Medium Story

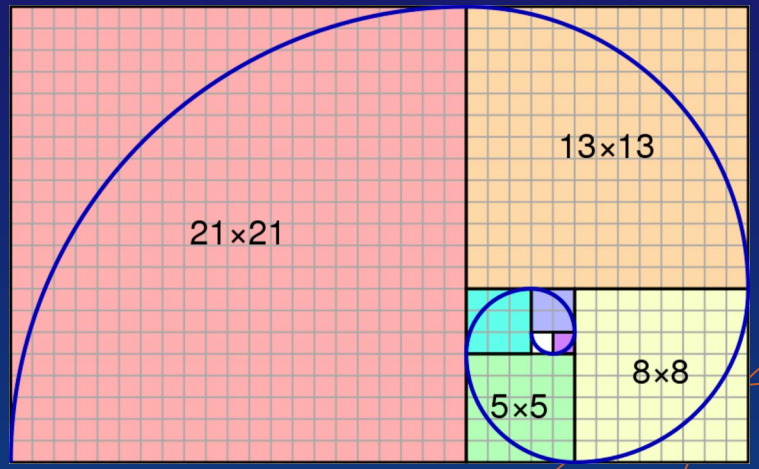


3 Story Points

Estimating Using Fibonacci Numbers

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711, 28657, 46368, 75025, 121393, 196418, 317811, …





1. **It's Composed of Integers**
2. **It's Non-linear**
3. **It Forces You To Choose "More Or Less"**
4. **It Sounds Cool And Adds An Air Of Legitimacy**

Estimating Using T-Shirt Sizes!

Point Values:

2	**5**	**15**	**25**	**70**	**150**

Estimating Considerations

Complexity	Uncertainty	Effort	Risks	Dependencies
How many components are involved?	Have we done this before?	How much work is involved? Can we reuse code or patterns?	Is there unproven techniques deployed?	Is there capabilities needed from a third party?
Are we using new technologies or skills?	Do we know who will be involved in the project?	How many people do we think we need?	Will we have a lot of new staff working on the project?	Do we have any serial dependencies?
How many stakeholders do we need to consider?	Do we have pre-existing guidelines or standards we can use?	What preparation is required?	Are there security, financial, or human risks inherent in the solution?	
Are there regulations to consider?	Do we have the skills needed on the team?	How much coordination/meetings will be needed?		

Game Time! Planing Poker!

https://planningpokeronline.com/

- Assign someone in your team to set up the game
  - Invite members of your team with link
- Use the T Shirt sizing estimate method
- Use the TigerChow Epics and Stories you just created.
  - Pick 5 Epics (across different user roles)
- Add the 5 Epics to the Issues in Planning Poker
  - (5 is the free limit)
- Play!

Teams in Software Development

The Importance of Teams in Software Development

Conway's Law

"Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure."

— Melvin E. Conway

Team = Software

"You can't have great software without a great team, and [many] software teams behave like dysfunctional families"

— Jim McCarthy

Important Lesson in Software Projects

- Jim McCarthy – Visual C++ director at Microsoft in 90's-2000s; considered one of the greatest teams ever at Microsoft
  - https://mccarthyshow.com/about/

- From his book Dynamics of Software Development
  - Don't Flip the Bozo Bit!

Don’t Flip the Bozo Bit

https://www.youtube.com/watch?v=QSKLm8E6KyE&list=PL9B1543FBFFB18EDD

Don’t Go Dark

https://www.youtube.com/watch?v=9OJ9hplU8XA&list=PL9B1543FBFFB18EDD

Beware of a Guy in A Room

https://www.youtube.com/watch?v=oY6BCHqEbyc&list=PL9B1543FBFFB18EDD

Discussion

At your tables…
5 minutes

- Each person in the team share the best team you were part of and the key characteristics that made the team great
  - Team can be school, sports, hobby, etc.
- Keep track of these characteristics
  - Make special note of any repeats
- Report on your “great” team characteristics to the class

TigerChow Project

CUSports Project

The team project for Spring 2025 is to create services/APIs and tests as required by the TigerChow application. The teams will develop services and their corresponding APIs to support a set of epics as specified by the Product Owner (Dr. Alex). Sprint management will be done in Trello. The API testing will be done in Postman using an API application built and deployed in AWS.

Technologies:

- Open API Specification
- JSON
- Postman/Newman
- Trello
- AWS: API Gateway, Lambda, Node/Python, DynamoDB

CUSports Project: Sprints

- Project is worth 350 points (35% of grade)
- All team members expected at sprint reviews
  - If not, will impact the absent student's grade
- There will be 4-5 total sprints
  - Including Sprint 0
- More details after the Kickoff Sprint (Sprint 0)

CUSport Project: Teams

- 4 members per team to keep teams “Agile-sized”
- Team members have been set up as Trello Workspaces and Canvas groups for grading
- Be sure to make sure you have access to Trello
  - (TA will make sure you have an invite)

Teamwork Survey & Grade

At the end of each sprint, your survey score will impact your grade as follows:

- 0.0 >= survey score < 1.0 for a Sprint – 0% for that sprint
- 1.0 >= survey score < 1.5 for a Sprint – 50% for that sprint
- 1.5 >= survey score < 2.0 for a Sprint – 60% for that sprint
- 2.0 >= survey score < 2.5 for a Sprint – 70% of team Sprint grades
- 2.5 >= survey score < 3.0 for a Sprint – 80% of team Sprint grades
- 3.0 >= survey score < 3.5 for a Sprint – 90% of team Sprint grades

If the entire team agrees that you were scored unfairly for a particular sprint (email sent copying each member of the team) - I will reconsider.

Teamwork Survey

TEAMWORK VALUE RUBRIC

for more information, please contact value@aacu.org

Definition

Teamwork is behaviors under the control of individual team members (effort they put into team tasks, their manner of interacting with others on team, and the quantity and quality of contributions they make to team discussions.)

Evaluators are encouraged to assign a zero to any work sample or collection of work that does not meet benchmark (cell one) level performance.

	Capstone 4	Milestones 3	Milestones 2	Benchmark 1
Contributes to team meetings	Helps the team move forward by articulating the merits of alternative ideas or proposals.	Offers alternative solutions or courses of action that build on the ideas of others.	Offers new suggestions to advance the work of the group.	Shares ideas but does not advance the work of the group.
Facilitates the contributions of team members	Engages team members in ways that facilitate their contributions to meetings by both constructively building upon or synthesizing the contributions of others as well as noticing when someone is not participating and inviting them to engage.	Engages team members in ways that facilitate their contributions to meetings by constructively building upon or synthesizing the contributions of others.	Engages team members in ways that facilitate their contributions to meetings by restating the views of other team members and/or asking questions for clarification.	Engages team members by taking turns and listening to others without interrupting.
Individual contributions outside of team meetings	Completes all assigned tasks by deadline; work accomplished is thorough, comprehensive and advances the project. Proactively helps other team members complete their assigned tasks to a similar level of excellence.	Completes all assigned tasks by deadline; work accomplished is thorough, comprehensive and advances the project.	Completes all assigned tasks by deadline; work accomplished advances the project.	Completes all assigned tasks by deadline.
Fosters constructive team climate	Supports a constructive team climate by doing all of the following: • Treats team members respectfully by being polite and constructive in communication. • Uses positive vocal or written tone, facial expressions, and/or body language to convey a positive attitude about the team and its work. • Motivates teammates by expressing confidence about the importance of the task and the team's ability to accomplish it. • Provides assistance and/or encouragement to team members.	Supports a constructive team climate by doing any three of the following: • Treats team members respectfully by being polite and constructive in communication. • Uses positive vocal or written tone, facial expressions, and/or body language to convey a positive attitude about the team and its work. • Motivates teammates by expressing confidence about the importance of the task and the team's ability to accomplish it. • Provides assistance and/or encouragement to team members.	Supports a constructive team climate by doing any two of the following: • Treats team members respectfully by being polite and constructive in communication. • Uses positive vocal or written tone, facial expressions, and/or body language to convey a positive attitude about the team and its work. • Motivates teammates by expressing confidence about the importance of the task and the team's ability to accomplish it. • Provides assistance and/or encouragement to team members.	Supports a constructive team climate by doing any one of the following: • Treats team members respectfully by being polite and constructive in communication. • Uses positive vocal or written tone, facial expressions, and/or body language to convey a positive attitude about the team and its work. • Motivates teammates by expressing confidence about the importance of the task and the team's ability to accomplish it. • Provides assistance and/or encouragement to team members.
Responds to conflict	Addresses destructive conflict directly and constructively, helping to manage/resolve it in a way that strengthens overall team cohesiveness and future effectiveness	Identifies and acknowledges conflict and stays engaged with it	Redirecting focus toward common ground, toward task at hand (away from conflict)	Passively accepts alternate viewpoints/ideas/opinions.

CUSports Project: Sprint 0

Sprint 0 (25 points):

- Team Kickoff and TigerChow epics using the Trello Boards in your workspace
- Use the Kickoff and Epic boards in your workspace
- 2-3 minute Sprint Review per team next Thursday 2/13 to present:
  - Team Kickoff Board with team name and logo
  - Epic and Services board

- **5 points** for thoughtful kickoff board/name/theme/logo
- **15 points** for thoughtful Epics, Estimates, Prioritization
- **5 points** for the Sprint Review
  - if you are not present you lose the points
- Team Survey due at end of day Friday 2/14
  - if you don't do the survey, it will impact your grade!

Up next…

- QUIZ 2 on Tuesday Feb 11
  - Weeks 3 - 5
  - Closed-note
  - 35 pts

- **Sprint 0 Review Thursday 2/13**

- **Microservices Assignment** - Details on Canvas