CPSC 3720 Lesson 7

User Stories Part 3

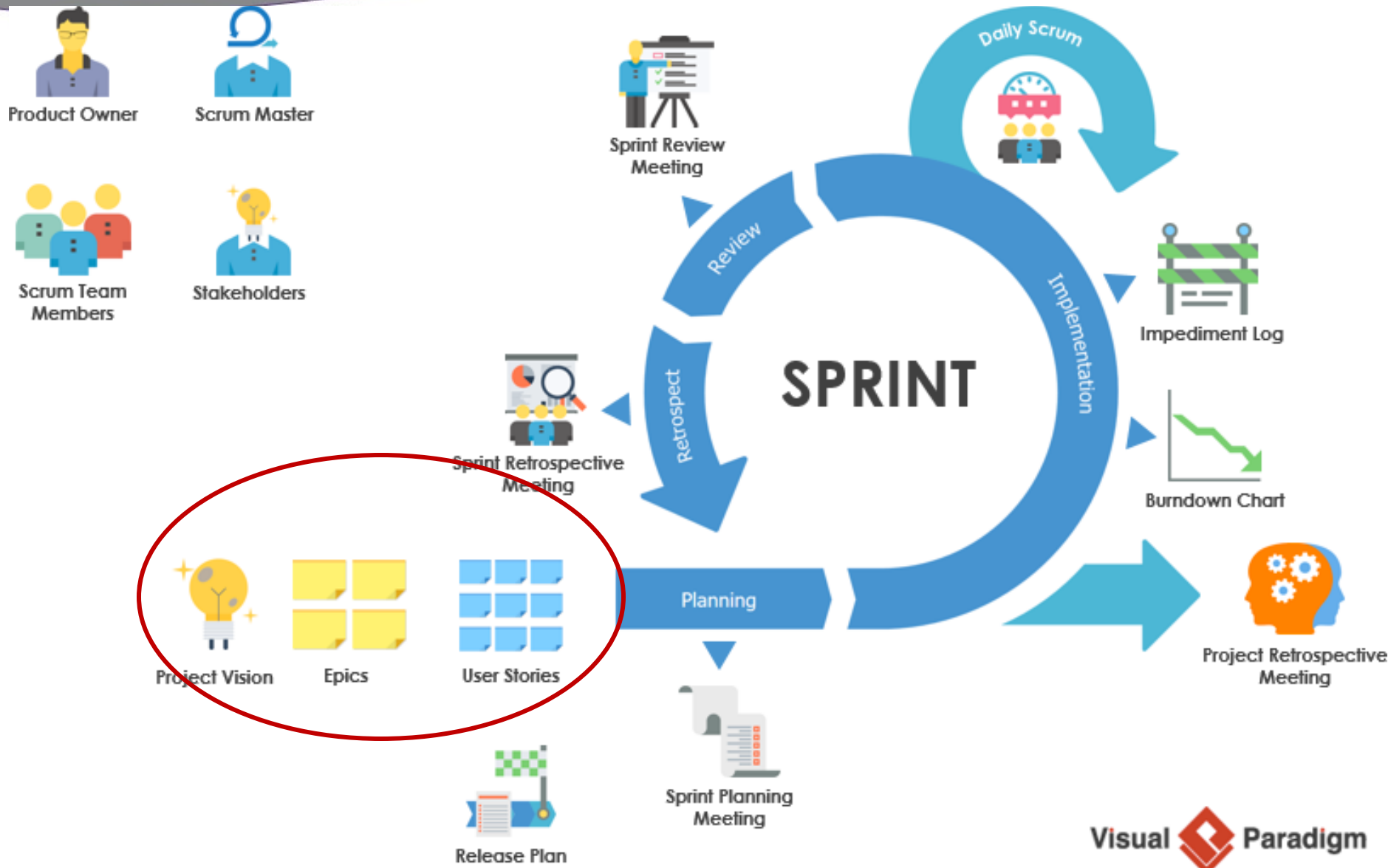**Connie Taylor**
Professor of Practice

School of **COMPUTING**

Today's Objective

- Creating User Stories – Recap

- Good Stories – Games

- Gathering Requirements

Scrum in 1 Picture

User stories are short, simple descriptions of a feature told from the perspective of the person who desires the new capability, usually a user or customer of the system. They typically follow a simple template:

As a ‹ type of user/user role › (WHO), I want ‹ some goal › (WHAT), so that ‹ some reason › (WHY)

The Three C's of
User Stories

Name the three Cs in story making

0 response submitted

Scan the QR or use link to join

https://forms.office.com/r/CjLY49hCjY

Copy link

Waiting for response...
Responses will be displayed in a list

Wordcloud | All responses

1 of 1

What is INVEST?

INVEST

S.A.M. – helps with testable
- Specific
- Attainable
- Measurable

Story Hierarchy

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

Stories should answer- Who, What, Why?

- Who – who 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)

- **4 Games**

- **Deck of cards separate for each game**

- **I will have a slide of instructions for each game**

Who, What, Why Game

- Who – who the user story is for
- What – the functionality that the user story implements
- Why – the reason the user needs the user story

- Every story needs a who, what, and why. Separate the W deck stories into those that are good stories or are missing one of the 3 W's.
- When you are done, turn over the cards to check your answers.
- You have 5 minutes.

<u>Sizing Game</u>

- Product – the largest chunk of value
- Epic – a piece of value that is larger than a story but smaller than the whole product
- User Story – a small piece of value that can be implemented in at most a week or two

- Separate the Z cards into the three sizes.
- When you are done, turn over the cards to check your answers.
- You have 5 minutes.

Negotiability Game

- The story is a conversation starter. There is a desired result but the exact method to obtain the result is open. Everything is user-oriented, not implementation-oriented. There are different levels of "negotiability".

- Order the N cards from most to least negotiable.
- When you are done, turn over the cards to check your answers.
- You have 5 minutes.

<u>Testability Game</u>

- The story provides enough clarity to determine how to objectively test the desired results.

- Separate the T3 cards into testable stories and not-testable stories.
- When you are done, turn over the cards to check your answers.
- You have 5 minutes.

Techniques for Gathering Stories

Questionnaires

Observation

User interviews

Story-writing workshops

How Do we Gather Requirements?

- ## The Waterfall way
  - Eliciting or capturing requirements
  - Implies users know and don't want to tell us

- ## The Agile way
  - Trawling* for requirements
  - Requirements change and mature or die
  - Can be caught with different size nets
  - Skill matters – this is not an easy task

*Mastering the Requirements Process by Suzanne and James Robertson, 1999.

Gathering User Stories

- Our "net" will not be so large or fine that we can get all user stores up front

- But we should come up with as many as possible

Techniques for Gathering Stories

Questionnaires

Observation

User interviews

Story-writing workshops

Gathering Stories

Questionnaires

- Good way to drill into existing stories
- If you have a large user base, questionnaires can help prioritize stories

Observation

- A terrific way to pick up insights on usage
- Can do this with or without a user's knowledge or through a user doing a demo

Gathering Stories

User Interviews

- Need to interview the right people and as many roles as possible
- Most users don't really know what they want (so don't just ask them what they want)

- Since users don't know how to solve their problems, we need to stop *asking*
- We need to *involve* them instead

Empirical design	▪ Designers of the new system make decisions by studying prospective users in typical situations
Participatory design	▪ The users of the system become part of the team designing the behavior of the system

User Interviews

Don't ask it this way:

"Would you like it in a browser?" — "Of course, now that you mention it!"

This is more open, so better. But, it has too much context.

"What would you think of having this app in a browser rather than as a native Windows application even if it means reduced performance, a poorer overall user experience, and less interactivity?"
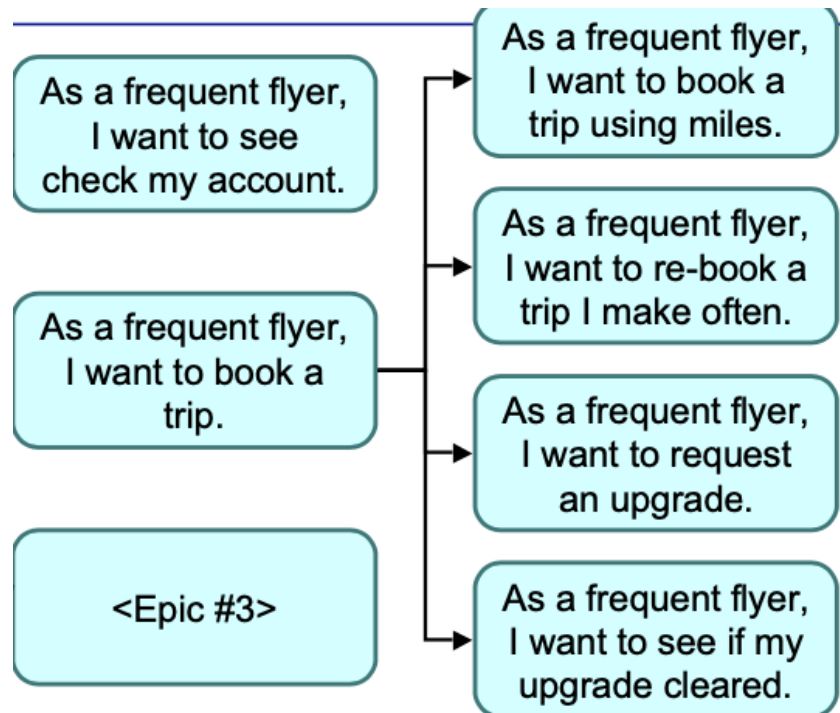
The best way
- Open-ended
- Context-free

"What would you be willing to give up in order to have it in a browser?"

Story-writing workshops

- Includes the developers, users, customers, others

- Write as many user stories as possible and they are not prioritized

- **Low-fidelity prototypes** and other brainstorming techniques can be used.

As a frequent flyer, I want to see check my account.

As a frequent flyer, I want to book a trip.

<Epic #3>

As a frequent flyer, I want to book a trip using miles.

As a frequent flyer, I want to re-book a trip I make often.

As a frequent flyer, I want to request an upgrade.

As a frequent flyer, I want to see if my upgrade cleared.

Low-fidelity prototyping

- Start with an empty box:
  - □ "Here's the main screen in the system"
- Ask open-ended, context-free questions as you go:
  - □ What will the users most likely want to do next?
  - □ What mistakes could the user make here?
  - □ What could confuse the user at this point?
  - □ What additional information could the user need?
- Consider these questions for each user role

Example: Low-fidelity prototype
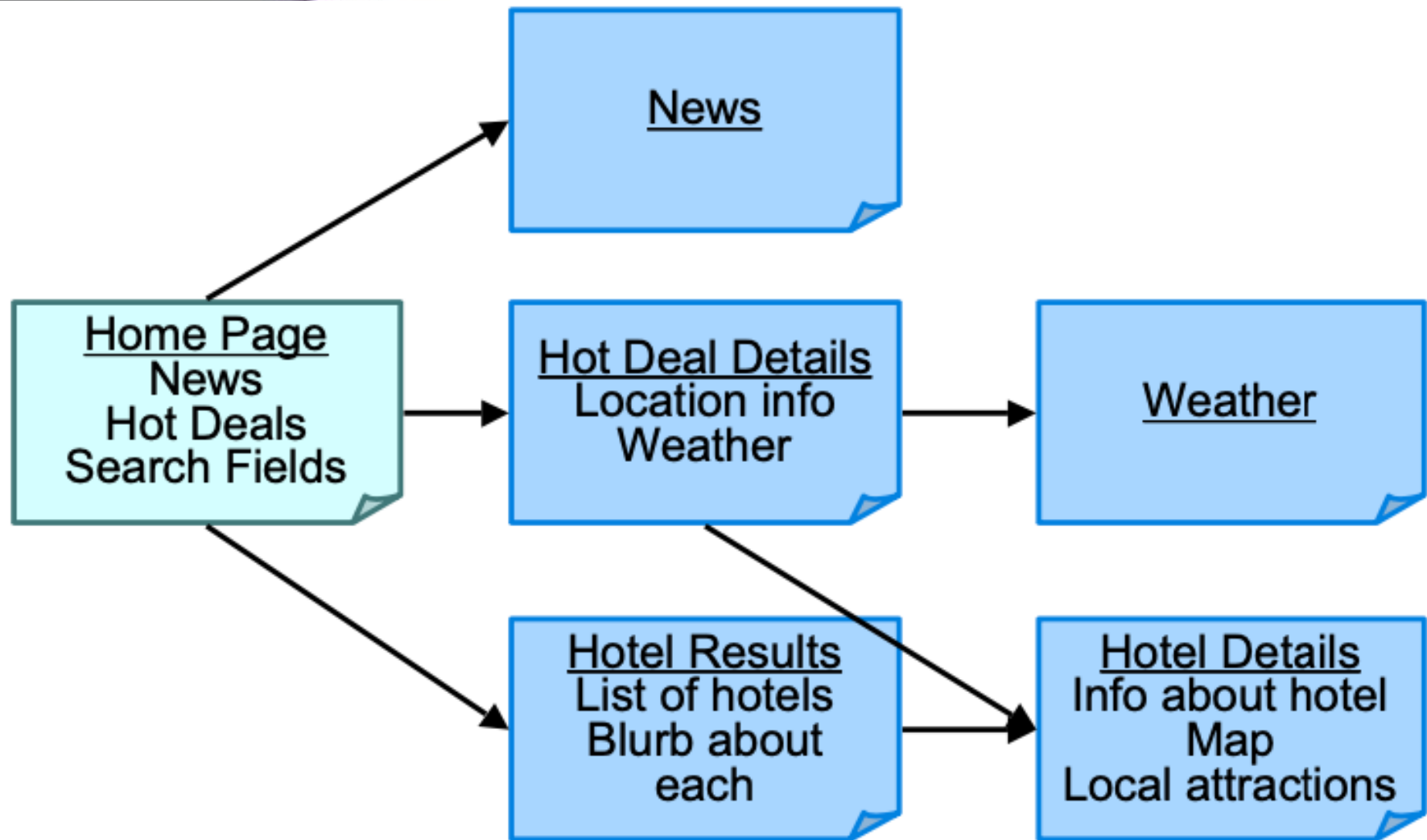
- **<u>Project Survey Link competed by this Friday at end of day</u>**

- Project Team Requests by end of day Friday via email (**<u>NO EXCEPTIONS</u>**)

- Project Kickoff next week

- QUIZ 2 on **Tuesday Feb 6** - will cover lessons 5-7; closed-note; 26 pts