Agile Requirements and Estimation

Because just coding didn't seem to work, did it?

Overview

- Effective Requirements
- User Stories
- Estimating Work
- Planning Poker

Effective Requirements

They exist.

What is a Requirement?

A Requirement is …

- A feature, behavior, or constraint to be added to a system
- A prelude to a conversation
- A request for someone to do work
- A request for software to change

What is a Requirement?

A Requirement is not …

- A solution design
- A decision about implementation
- Typically illustrative of the final deliverable
- The source of truth

NASA Requirements

The desirable characteristics for requirements specifications are:

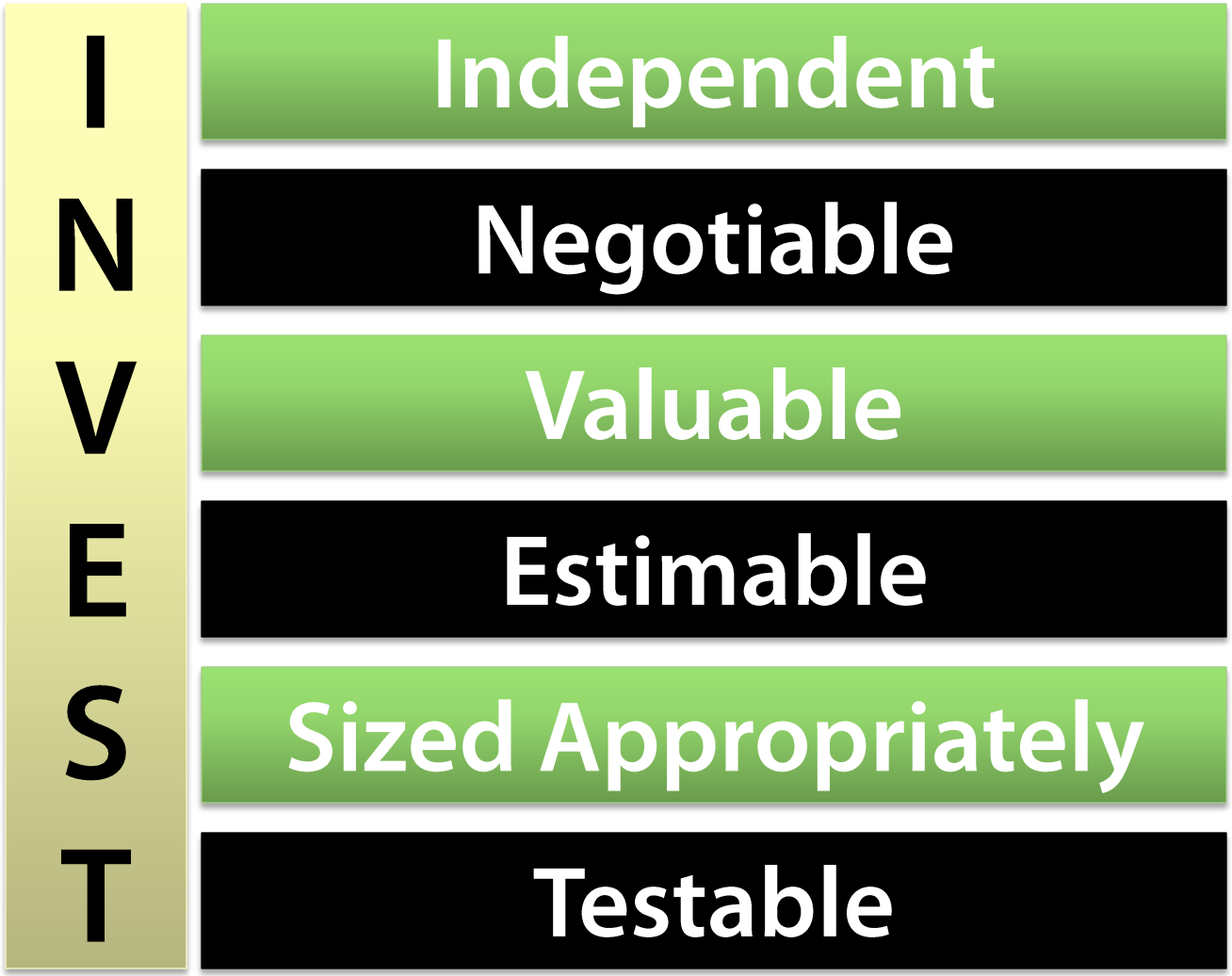
- Complete
- Consistent
- Correct
- Modifiable
- Ranked
- Traceable
- Unambiguous
- Verifiable

IEEE Requirements

IEEE says these things need to be present in requirements

- Interfaces
- Functional Capabilities
- Performance Levels
- Data Structures/Elements
- Safety
- Reliability
- Security/Privacy
- Quality
- Constraints & limitations

A Simple Work Item Recipe



"The significant problems we face cannot be solved by the same level of thinking that created them."

- Albert Einstein

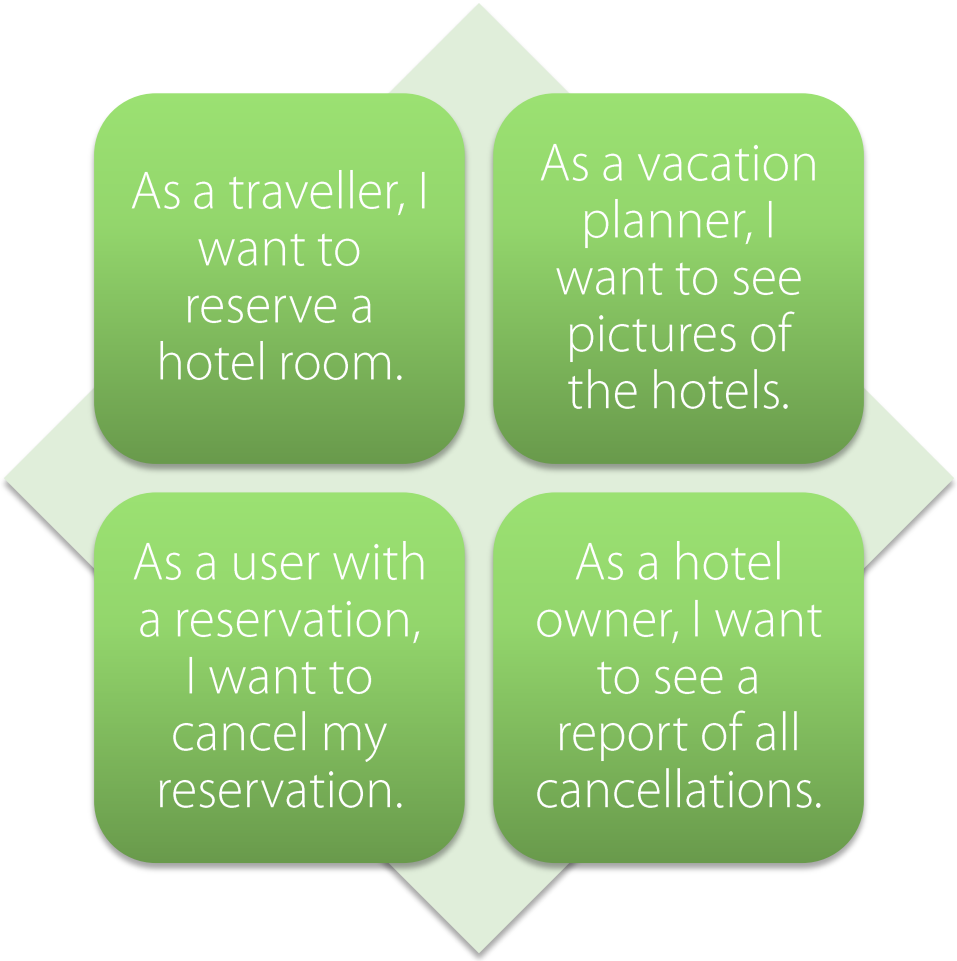
User Stories

The Currency of the All Work

User Story Recipe

As a <role> I want <feature> so that <benefit>.

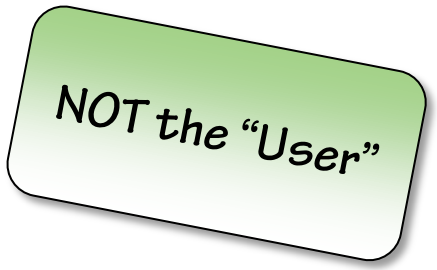
Some User Stories



A Note on Roles

As a <role>

- Vacationer
- Hotel Owner
- Travel Agent
- Trip Planner
- Parent



Why User Stories Work Well

- They are simple to write and understand
- Software requirements is a communication problem
- They elicit detail in conversations
- Requirements analysis is effective when performed collaboratively
- Full intent can rarely be modeled or represented 100%

The User Story Conversation

> As a user with a reservation,
> I want to cancel my reservation
> So that I get a refund.

- **Does the user get a full or partial refund?**
  - Is the refund to her credit card or is it site credit?
- **How far ahead must the reservation be cancelled?**
  - Is that the same for all hotels?
  - For all site visitors?
    - Can frequent travelers cancel later?
- **Is a confirmation provided to the user? How?**

Details as Smaller Sub-Stories

As a user with a reservation, **I want** to cancel my reservation **so that** I get a refund.

- **As a** premium member, **I want** to cancel at the last minute with no penalty **so that** I get a full refund.
- **As a** non-premium member, I want cancel up to 24 hours in advance **so that** I get a 50% refund.
- **As a** site member, **I want** an email confirmation of my cancelled reservation **so that** I can have a record of the transaction.

Signs Stories are Working

- Focus shifts from writing to talking
- Stories are understood by customer and developer
- At estimation time, they are the right size
- Participative design is occurring
- Emphasis is on the users goals, not the system's attributes

Communicating Done

Given <context> [and <more context>],

When <something happens>

Then <outcome> [and <another outcome>].

Given / When / Then Criteria

As a user with a reservation, **I want** to cancel my reservation **so that** I get a refund.

Given I am a premium member, **when** I cancel under 24 hours, **then** I incur no penalty.

Given I am a non-premium member, **when** I cancel less than 24 hours in advance, **then** I pay 50% fee.

Given I am a site member, **when** I cancel my reservation, **then** I am emailed a confirmation.

Adding Snappy Titles

Title: User cancels reservation

As a user with a reservation, **I want** to cancel my reservation **so that** I get a refund.

Scenario 1: User is a premium member

Given I am a premium member, **when** I cancel under 24 hours, **then** I incur no penalty.

Scenario 2: User is a typical member

Given I am a non-premium member, **when** I cancel less than 24 hours in advance, **then** I pay 50% fee.

Scenario 3: User gets an email confirmation

Given I am a site member, **when** I cancel my reservation, **then** I am emailed a confirmation.

The Whole Story on a Card

Title: *User cancels reservation*

Description

As a user with a reservation, I want to cancel my reservation so that I get a refund.

Success Criteria

- *Given I am a premium member, when I cancel under 24 hours, then I incur no penalty.*
- *Given I am a non-premium member, when I cancel less than 24 hours in advance, then I pay 50% fee.*
- *Given I am a site member, when I cancel my reservation, then I am emailed a confirmation.*

Business Value Estimate _____
Development Effort Estimate _____
ROI Estimate _____

Story Owner _____

It is very difficult to make a vigorous, plausible, and job-risking defense of an estimate that is derived by no quantitative method, supported by little data, and certified chiefly by the hunches of the managers.

— Fred Brooks (1975)

Estimating Work

Without dartboards, guesses, or making it up

Estimates are Necessary

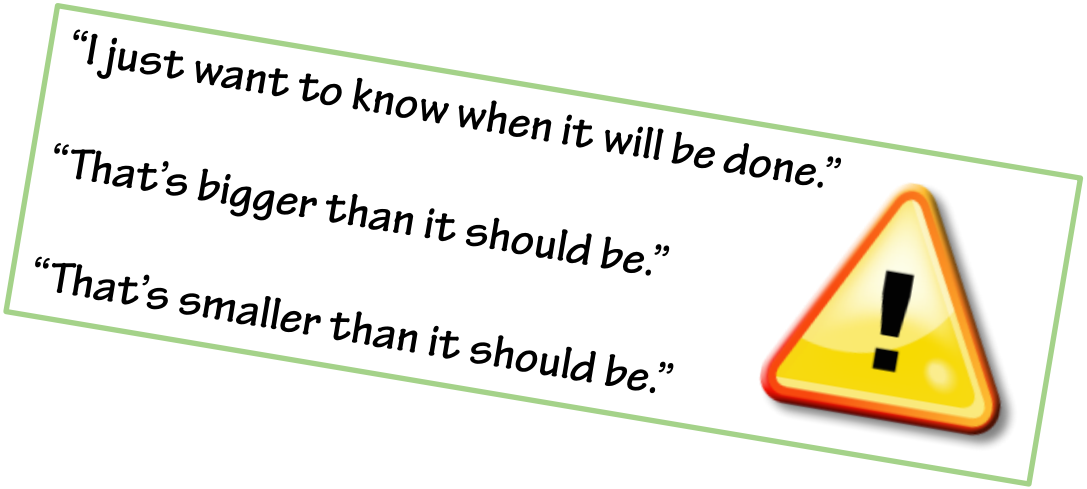
- To plan and proceed deliberately
- To get a feel for costs
- To calculate potential ROI
- To understand the size of something
- To know if work even can be done
- To weigh options

Ways to Estimate Software

- Darts
- Give it to the manager
- Ask the expert
- Without “bothering” the developers. They’re busy.

Deadly Estimation Warning Signs

- Someone other than the team is doing the estimation.
- Estimates are given without looking at historical performance.
- Estimates are treated as promises
- Estimates are rejected because they don't fit an already existing plan



The Typical Estimation Process

PM: *Hey, Bill, how long to _____?*

Dev to self: *I'm busy, but he's back. That'll Take 2 days I can't afford to lose. What can I say that will make him go away?*

Dev out loud: *About a week.*

PM to self: *They always say that. So, 2.5 weeks. I'll make it 3.*

PM out loud: *Thanks, Bill. I'll go write the specs now.*

Dev to self: *I can stall those out for weeks.*

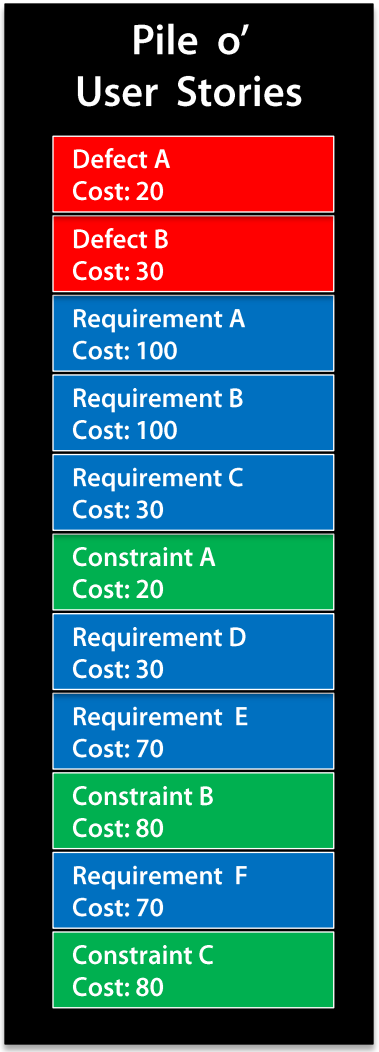
How do we measure software work?

Story Points

- Very common way to estimate work
- Based on size and complexity, not duration
- Unitless and numerically relative
- Different for each team of estimators
- Points are additive, unlike time
- Based on historical reality
- Easy to use and understand

Acorns?

Using Story Points



We can see right away

1. Which work items cost the most
2. Total cost of all the work
3. Total cost to an iteration

Story Point Values

- **Can you distinguish a 1-point story from a 2?**
- **Can you distinguish a 17 from an 18?**
- **How about a 99 from a 100?**
- **Use units that make sense**
  - XS, S, M, L XL, XXL
  - 1, 2, 3, 5, 8, 13, 20, 40
  - 1, 2, 4, 8, 16, 32

Include big and small outliers if you want.
0, ½, 100, 300, ∞

It's Called an Estimate

Not a promise. Don't worry so much.

Remember why estimates are needed

- Large scale planning
- Get a feel for cost

If estimates are used against you, this is a people problem, not a problem with the estimates.

Address it.

The Starr kids
estimating their chores.

Planning Poker

It's what you think.

Estimating with Groups

- **Group derived estimates are demonstrably more accurate than estimates by individuals**

> Together, we are smarter than any one of us.
> – Japanese proverb

- **Political Trading Markets**
  - Iowa Electronic Markets
  - Intrade.com
  - politicalmarket.cnn.com

- **"Who wants to be a Millionaire?" Polling the audience is accurate 91% of the time.**

> When guessing the number of jellybeans any given jar, the average of all guesses is typically within 2-3% of the correct answer.
>
> - The Wisdom of Crowds, James Surowiecki

Relative Estimation

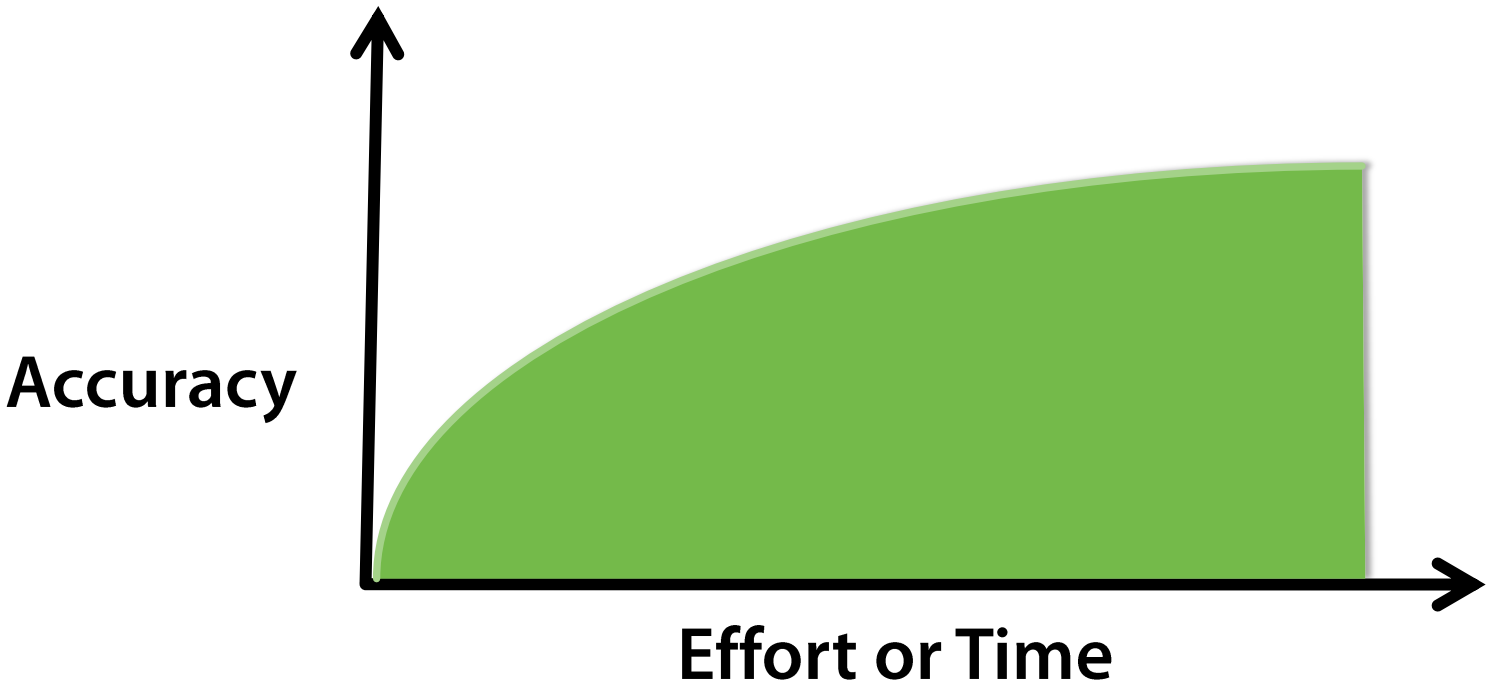
A

B

Myth

With more time, estimates get significantly more accurate.

Estimation is Expensive



Planning Poker Cards

Why Planning Poker Works

- Emphasizes relative sizing
- Focuses most estimates within an order of magnitude
- Everyone is heard
- Finds hidden requirements and details
- Estimators must justify estimates
- It is iterative
- It's quick
- It's fun

Planning Poker Rules

1. Each estimator has a deck of cards estimation cards.
2. Customer/Product Owner reads a story and it's discussed briefly.
3. Each estimator selects a card that's his or her estimate.
4. Cards are turned over so all can see them (synchronously).
5. Discuss differences (especially outliers).
6. Re-estimate until estimates converge.

A Real Work Item

Check table widths before Checkin or Save

Before a user saves or checks in a document, test all of the tables in the document to see if they follow the XHTML rules. If they don't, throw a warning to the user.

Don't throw an error.

To Be Done:

```
for each table:
If ( table.width == 100% )
        Throw an error
Else If( table.width <
100opixels)
        throw an error
Else pass
```

PTS

Planning Poker

	Round 1
Homer	8
Marge	5
Bart	1
Lisa	5
Maggie	3

Planning Poker

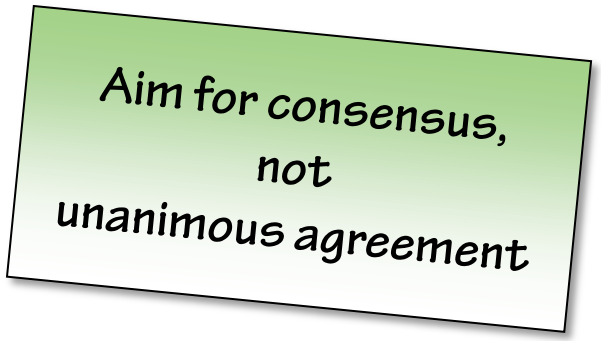
	Round 1	Round 2
Homer	8	13
Marge	5	5
Bart	1	3
Lisa	5	5
Maggie	3	5

Planning Poker

	Round 1	Round 2	Round 3
Homer	8	13	1
Marge	5	5	5
Bart	1	3	5
Lisa	5	5	5
Maggie	3	5	5

Options for Handling Conflict

1. Wait for convergence
2. Average the estimates
3. Toss out high and low
4. Send the item back for re-definition



Planning Poker Workshop

Give it a whirl.

Make A Planning Poker Deck

?, 1, 2, 3, 5, 8,

13, 20, 40, ∞

Try These

Backlog Item	Estimate
Mow my lawn	5
Move your slacker friend from his mom's house to an apartment	
Paint my house	
Write Pong in Silverlight	
Add a new team member	
Make 8 pounds of confetti	

Estimation Workshop

The Basic Idea

1. Break into groups
2. Estimate the provided work items
3. Stop periodically to check in with the whole group

Summary

- **Effective Requirements**
  - Remember INVEST
- **User Stories**
  - As a <role> I want <feature> so that <benefit>.
  - Given <context> when <something happens> then <desired result>.
- **Estimating Work**
  - Relative estimation
  - Group estimation
- **Planning Poker**

References

- ***Writing Effective Requirements Specifications***
  satc.gsfc.nasa.gov/support/STC_APR97/write/writert.html
- ***Blink: The Power of Thinking Without Thinking,*** **Malcolm Gladwell**
- ***The Wisdom of Crowds*****, James Surowiecki**
- ***Agile Estimating and Planning,*** **Mike Cohn**
- ***User Stories Applied For Agile Software Development,*** **Mike Cohn**
- *PlanningPoker.com*