Overview

→ Agile 101
→ Purpose of User Story
→ Three C’s and three V’s
→ Basic Template Elements: who, what, why
→ Acceptance Criteria and Testing Notes
→ Simplifying Stories
→ Epics, Spikes, Tracer Bullets
Agile Manifesto 101

Individuals & Interactions
Working Software
Collaboration with Customer
Responding to Change

Process & Tools
Comprehensive Documentation
Contract Negotiation
Following a Plan
Why do we need User Stories?

→ Define Business value
→ Work within current conditions only
→ Avoid clarity and completeness
→ Divide work into digestible chunks
User Story Purpose – Kent Beck

User stories are called user stories because this is how they are supposed to be used.

→ Tell a story about what software does.
→ Generate a productive discussion with interest and vision
→ Agree on how to know when software is done
User Story Process – Ron Jeffries

Cards

- For each thing you want software to do, write that down on an index card
→ Conversation

- Reconcile what the person writing the story and the person building it both understand
- Discuss to discover different interpretations
- Clarify and refine the story
- Discuss to explore solution scenarios
Once it is built, how do you check that it’s done?

- Check against a list of things (acceptance criteria)
- Demonstrate functionality
How to build a Minimum Viable Product

Not Like This:

1  2  3  4

Like This:

1  2  3  4  5
CROSSED WIRES
Product Owners

→ VISION – Define the goals for the product
→ VALUE – What is it worth to the business
→ VERBAGE – Business language and logic
Template Zombies
The project team allows its work to be driven by template instead of by the thought process necessary to deliver products.

The value of writing user stories is not what’s written down, it is what we learn as we tell the story.

Think about the core value the story helps to achieve
Best Practices

→ Don’t be a template zombie – think about the story, not the template
→ Personas
→ Define goals clearly
→ Simplify the story

Let’s look at examples!
As a user I want the site to use minified .js
User Story Template

WHO

→ Personas - not just a ‘user’ but a broader kind of user
   ▪ Business customer, Stakeholders
   ▪ Anonymous users, authenticated users
   ▪ Specific use case if it helps to explain the story better
   ▪ Content editor, content creator
User Story Template

WHAT

→ Explains the desired end result or the task to be accomplished
→ Basic tasks
→ Underlying systems and services
→ UI and UX
→ Understanding process
User Story Template

WHY (Who cares?)
→ Why should a user care?
→ Why does the company care?
→ Why should other users care?
→ Why should stakeholders care?
→ Align the story with a core theme based on business model and process – what does this help achieve?
User Story Template

Other things to think about:
→ What happens when things go wrong?
→ Question your assumptions
→ Question technical assumptions
Writing better user stories

→ **Who:** Using personas instead of “user”
→ **What:** Clearly define stories
→ **Why:** Use relevant drivers that everyone can relate to
→ **What is the reason** for this story? The core purpose?

As a user, I want the site to use minified js.
Define the goal of the story

→ The WHAT of the story should define the goal, not the solution.

→ Clearly understood definition of done.

As a fan of customer.com, I want the site to use minified js to load faster.
Define the goal of the story

→ The WHAT of the story should define the goal, not the solution.

→ Clearly understood definition of done.

As a fan of customer.com, I want the site the home page to use minified js to load in under 5 seconds.
Writing clear user stories

→ Avoid the use of ‘and,’ ‘or,’ ‘but’

→ Avoid the use of limiters: ‘unless,’ ‘until,’ ‘without,’ and ‘except’

As a fan of customer.com,
I want the home page to load in under 5 seconds unless on a mobile device.
Explain in simplest terms

→ Learn and understand the core business drivers:
  - Revenue
  - Quality
  - Customer Experience
  - Other Business Goals

As a fan of customer.com, I want the home page to load in under 5 seconds so that I’m not easily distracted and leave the site.
So many acronyms

SMART – Acronym #1

→ Specific?
→ Measureable?
→ Achieveable?
→ Relevant?
→ Time-boxed?

As a fan of customer.com,
I want the home page to load in under 5 seconds so that I’m not easily distracted and leave the site.
So many acronyms

INVEST – Acronym #2
→ Independent
→ Negotiable
→ Valuable
→ Estimable
→ Sizeable
→ Testable

As a fan of customer.com, I want the home page to load in under 5 seconds so that I’m not easily distracted and leave the site.
More Examples

→ As a user, I want the site to look like the attached .psd file.

→ As a Publisher or Reviewer I want the ability to set specific days and times to publish or expire (unpublish) content so that I can automate publishing tasks of time-sensitive content.
Acceptance Criteria

→ Easiest format to remember:
→ Given that [someone wants to do something]
→ When s/he [performs action]
→ Then [something happens that can be verified]
Testing Notes

→ How will you validate what was done?
→ As an anonymous user/content admin
→ I visit page, click on link, something happens.
→ Behat Test Writing!
Other Tips & Tricks

→ Summary: short & sweet
  ▪ Instant Release Notes
→ Labels: another way to organize Stories
  ▪ Broader themes or groupings
→ Priorities:
  Blocker, Critical, Major, Minor, Trivial
Reducing Stories Down

→ Consider CRUD
  ▪ Create, Read, Update, Delete
→ Consider Personas
  ▪ Users, Content Editors, Administrators
→ Consider Business Value
  ▪ Logic, Added Value, New Product
EPICS
Epic Organization

→ Section of a site
→ Loose group of associated tickets
→ Key deliverable in a site
Epic Organization

→ Theming
  ▪ New Subtheme
→ Integrations
  ▪ Integration – Salesforce
→ Functions
  ▪ Function – User Profile
→ Setup
  ▪ Setup – Automated Testing
→ UX
  ▪ UX – Global Navigation

→ News Releases
→ Front Page
→ Photo Gallery
→ Contact Form 1.0
→ Events Calendar 1.1
→ Product Page 2.0
SPIKES
Best Practices: Spikes

- Research task
- Usually timeboxed
- Answers if something is possible
- Not deliverable code, but throwaway work
TRACER BULLET
Best Practices: Tracer Bullet

→ Narrow slice through layers of an epic
→ Single focused goal
→ Solve key architecture or design issues
→ Could be [the basis of] deliverable code
ONE MORE THING
Sub-Tasks

→ Useful to help keep track of process or workflow-related issues
  - Tracking external review teams
  - Tracking priorities among tickets
  - Announce need for a check-in meeting
  - Reminder to create QA test script
JIRA Components

→ Associate with content types “Content: Article”
→ Organizes tickets to optimize traceability
→ Subset QA Testing
→ Utilize general use cases “Content Editing,” “Image Editing”
→ Add others based on Epics “Analytics,” “Advertising,” “Metadata,” “Global Menu,” “Footer”
What You Learned

→ Agile Manifesto
→ Not to be a Zombie
→ User Stories: 3Cs and 3Vs
→ User Stories: SMART INVESTING
→ How to Simplify & Clarify
→ Personas
→ Epics
→ Spikes
→ Tracer Bullets
→ Use Components
→ Sub-Tasks
→ Other tips & tricks
Questions?

Chris Urban, Engagement Manager
chris.urban@acquia.com
@_urban_  d.o:urbanlegend