Introduction to Agile Software Development Practices

Dev Barua, CSM, CSPO, PMP
Sr. Development Manager
Scope of the presentation

- Why organizations fail to adopt the agile development process
- Introduction to Agile Development
- Types of agile methods and an introduction to scrum
- How the agile-scrum process has been adopted at HelpSystems
Top 3 Reasons Companies Struggle with Agile and Scrum
Top Three Reasons

- Organization Culture
- Lack of Understanding of What Agile/Scrum Means
- Not “Using As Directed”
Introduction to Agile
What is Agile?

Agile is about enabling business results

- An approach to product development
- Adaptive – there is no “Specific or The Agile Method”
- To “be agile” put the values and principles into practice
- Processes such as Scrum and eXtreme Programming (XP) are considered “Agile” because they adhere to Agile’s Values & Principles
Agile Methods

Extreme Programming (XP)  Scrum  Lean Software Development  Kanban  DSDM  Crystal Methodology  FDD
The Agile Manifesto—a statement of values

- **Individuals and interactions**
- **Working software**
- **Customer collaboration**
- **Responding to change**

- **Process and tools**
- **Comprehensive Documentation**
- **Contract negotiation**
- **Following a plan**

Source: www.agilemanifesto.org
Introduction to Scrum
Introduction to Scrum

It is...

• A simple **framework** that can be understood and implemented in short time
• An approach to **managing complexity**
• A **collaborative** effort that enables an engaged **ongoing dialog**
• **Most popular** Agile Method used today
• Has **industry supported** standards (roles, tools, certification, etc.)
• A culture change for the **entire organization**

It is not...

• A methodology
• A license to do **NO documentation** (some documents still need to be created)
• A silver bullet for all that **ails software development**
• A framework which provides **detailed plans** for every contingency
Mindset of SCRUM?

In the figure above, these three things depict a simple equation:

**People use Practices to develop Product**

*Source: Exploring Scrum by Dan Rawsthorne with Doug Shimp*
Scrum Theory

Three pillars uphold every implementation of empirical process control: transparency, inspection, and adaptation.

- **Transparency**: Significant aspects of the Scrum process must be visible to those responsible for the outcome, i.e. the scrum team.

- **Inspection**: Scrum users must frequently inspect Scrum artifacts and progress toward a Sprint Goal to detect undesirable variances.

- **Adaptation**: If the Scrum determines that one or more aspects of a process deviate outside acceptable limits, the process must be adjusted.

*Source: Scrum Guide by Ken Schwaber and Jeff Sutherland*
Scrum Process
SCRUM Development Process

- **VISION**
  - Vision Roadmap

- **REFINEMENT**
  - Decide Definition of Done

- **RELEASE PLANNING**
  - Sprint Backlog

- **SPRINT PLANNING**
  - 2-4 Week Sprint
    - 15 Minutes Daily SCRUM Stand up
    - Potentially Shippable Product Increment

- **SPRINT REVIEW**
  - Sprint Retrospective

- **REPEAT...**
## Product Backlog Example

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Priority</th>
<th>Type</th>
<th>Estimate</th>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A Developer can make requests to a public WebDocs interface to retrieve general information about WebDocs</td>
<td>M</td>
<td>Design</td>
<td>2</td>
<td>1.8</td>
</tr>
<tr>
<td>2</td>
<td>A Developer can make requests to a public WebDocs interface so that folder information can be manipulated</td>
<td>M</td>
<td>Code</td>
<td>3</td>
<td>1.7</td>
</tr>
<tr>
<td>3</td>
<td>A Developer can make requests to a public WebDocs interface so that user information can be manipulated</td>
<td>S</td>
<td>Asset</td>
<td>8</td>
<td>1.8</td>
</tr>
<tr>
<td>4</td>
<td>A Developer can make requests to a public WebDocs interface so that group information can be manipulated</td>
<td>W</td>
<td>Design</td>
<td>8</td>
<td>1.8</td>
</tr>
<tr>
<td>5</td>
<td>A Developer can make requests to a public WebDocs interface so that permission information can be manipulated</td>
<td>M</td>
<td>Code (popup)</td>
<td>2</td>
<td>1.8</td>
</tr>
<tr>
<td>6</td>
<td>A user can view the details of a document so that the document metadata is available</td>
<td>M</td>
<td>Code/Asset</td>
<td>1</td>
<td>1.8</td>
</tr>
<tr>
<td>7</td>
<td>A user can use the main application menu in order to navigate to the primary functional areas of the interface</td>
<td>S</td>
<td>Code</td>
<td>3</td>
<td>1.8</td>
</tr>
</tbody>
</table>

Source: RJS Software, HelpSystems

<table>
<thead>
<tr>
<th>ID</th>
<th>Product Backlog Item</th>
<th>Order</th>
<th>Acceptance Criteria (Where Applicable)</th>
</tr>
</thead>
</table>
| A  | As a parent, I want hot dogs so that my kids have something to eat at the picnic.    | 1     | • There must be enough hot dogs to feed 10 children.  
• All hot dogs are free of additives.                                     |
| C  | As a picnic attendee, I want hamburgers because nothing tastes better at a summer picnic than a hamburger | 2     | • There must be enough hamburgers to feed 10 adults.  
• All hamburgers must be made of ground beef or turkey.                    |
| L  | Water                                                                               | 3     | Not Applicable                                                                                       |
| E  | As a picnic attendee, I want potato salad to accompany my main dish (either hamburgers or hot dogs) because potato salad enhances the flavor of both. | 4     | • Only Idaho potatoes should be used in the salad  
• Only Low-fat mayonnaise                                               |
| G  | Potato Chips                                                                        | 5     | Not Applicable                                                                                       |
| P  | Cupcakes                                                                            | 6     | Not Applicable                                                                                       |

Source: www.collaborativeleadershipteam.com
Product Backlog - Order/Re-order methods

- High Risk
  - Low Value
- Low Risk
  - Low Value
- High Risk
  - High Value
- Low Risk
  - High Value

Do last if we do these at all!!!!
Elements of SCRUM - $3^3$

**PEOPLE - 3**
- Scrum Team
- Product owner
- Scrum Master

**Ceremonies - 3**
- Sprint planning
- Daily stand-up
- Sprint Review

**Artifacts - 3**
- Product backlog
- Sprint backlog
- Burn down charts
Scrum Team – who are part of it?

- Business/Systems Analysts
- Developers
- QAs
- DBAs
- UI/UX Designers

- Anyone not in an authoritative role.
- An authority on the process but not the team’s boss
- Drives the process
- Is part of the team
- Should have some decision making authority
- Should not be the Scrum Master

The Scrum Team
Product owner

- Team Member who is accountable for the Business for the value of the Team’s Work Results
- Be the primary interface with the Stakeholders, and can be a Subject Matter Expert (SME)
- Have a vision for the Product, its Releases, and the Sprints
- Responsible for maximizing the value of the product; define and prioritize the features of the product according to market value
  - Can change features and priority every X days
- Maintain and groom a prioritized Backlog of Product Backlog Items on regular basis
- Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next;
- Determine what product the team will build, and drive the team as a sustainable pace
- Guides product development
  - Adjust features and priority every Sprint, as needed
- Seek guidance from the development team
- Has final say on the work product; can accept or reject work results (Sprint)
Product Owner in Every Step of SCRUM

- Be the primary interface with the Stakeholders, and can be a **Subject Matter Expert (SME)**
- Acting as **Product Owner** during Spring Planning, Sprint Demo and Backlog Grooming
- Acting as **Team Member** during Sprint Cycle.
The Scrum Master

- Represents management to the project
- Responsible for ensuring Scrum is understood and enacted
- Scrum Masters do this by ensuring that the Scrum Team adheres to Scrum theory, practices, and rules
- Removes impediments
- Ensure that the team is fully functional and productive
- Enable close cooperation across all roles and functions
  - Manage Scrum Team’s relationship with the Product Owner
- Shield the team from external interferences

A Scrum Master Can be from any number of team roles: Project Manager, Team Lead, Dev Lead, QA, etc.
Elements of SCRUM

Roles - 3
- Product owner
- Scrum Master
- Team

Ceremonies - 3
- Sprint planning
- Daily stand-up
- Sprint Review

Artifacts - 3
- Product backlog
- Sprint backlog
- Burn down charts
Agile Ceremonies

- **Sprint (or Iteration) Planning**: Held at the beginning of each sprint for the team so commit Product Backlog Items to the Sprint Backlog

- **Daily Stand up**: a 15 minute time-boxed event for the Team to inspect, adapt and transparently synchronize on the Sprint Goal

- **Sprint Review (through Demo)**: Feedback mechanism for stakeholders to see working product increments and for the Product Owner to inspect and adapt on the Product
  - **Sprint Retrospective**: an inspect and adapt mechanism for the Development Team regarding their Process

- **Release Planning**: a session that comes from the eXtreme Programming framework
  - A Release is typically made up of multiple Sprints or Iterations
Elements of SCRUM

Roles - 3
- Product owner
- Scrum Master
- Team

Ceremonies - 3
- Sprint planning
- Daily stand-up
- Sprint Review

Artifacts - 3
- Product backlog
- Sprint backlog
- Burn down charts
Product Backlog

• SINGLE SOURCE OF REQUIREMENTS and TRUTH
• An ordered list of everything that potentially be in the Product
  – All features, functions, requirements, enhancements, and fixes
• Is never complete; constantly changes to identify product needs
• Product Owner owns the product backlog
• The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering
  – Frequently re-ordered/re-prioritized
• Reprioritized at the start of each Sprint
How to breakdown backlog items

Epic
(A BIG Story)

Theme
(Collection of Related Stories)

Story
(Requirements from User’s Perspective)

Task

Task

Task

Actual work needed for Team to complete

Frequent Flyer

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

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

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

As a frequent flyer, I want to rebook a trip I take often.

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

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

Example:
How to write user stories

• User Stories typically follow a simple template:

    As a <type of user>, I want to <goal>, so that <reason>

Example 1: As a HR Admin, I want to see all of the time off requests in a queue, so that I can approve those in the order that I have received them.

• “If we work on the back-end of the software development, how can I write a user story?”

Example 2: As the document imaging system, I want to receive all data as valid, well-formed XML so that I don’t have to worry about syntax checking.

• If the syntax doesn’t fit any of the above mentioned format, write the backlog with syntax, used in Feature Driven Development

    <action> <result><object>

Example 3: Display mail icon on the queue

    Add delete button on each queue
Work Breakdown Method – Vertical Slicing

- User Story 1
- User Story 2
- User Story 3

Client Application
Web Interface
Business Logic
Database
Criteria for good user stories: INVEST

**Independent**
User stories should be independent of each other during their execution. Dependencies lead to problems for estimating and prioritizing.

**Negotiable**
Stories are not contracts; they leave or imply some flexibility. Stories are the negotiations units in Scrum and are agreed to in planning and are delivered.

**Valuable**
Stories are, by definition, unites of value that are requested by the users and customers.

**Estimable**
Team should be able to agree to the user stories for the correct estimates. If the team cannot estimate the stories, then the stories are ambiguous.

**Sized Appropriately**
Stories should be small enough to be completed in one sprint (Iteration). It is better to have a single focus per user story.

**Testable**
If the requirements are not testable, requirements are vague. Likewise, each story needs to be verifiable, so that the team can determine when it is done.

“Done” or “Not Done”, no “partially finished” or “done except”

Source: 3Back.com
Estimates of Work through User Stories

• For estimation, Story Points for user stories are widely used; T-Shirt size is also used by some scrum teams (Small, Medium, Large, X-Large), calendar days, and weeks.

• Story points indicate the size and complexity of the story relative to other stories.
  – A story with two story points is expected to take twice as long as a story with one point.

• Story Point determined by the team
  – No averages!
  – Make sure everyone is involved, team consensus.

• Select a pointing system, for example
  – Fibonacci sequence: 1, 2, 3, 5, 8, 13, 21+
  – Utilize a process such ‘planning poker”

• Set smallest story to a “1” and largest story to your highest number to establish markers.
Definition - Velocity

- Velocity is how much product backlog effort a team can handle in one sprint. This can be estimated by reviewing previous sprints, assuming the team composition and sprint duration are kept constant. It can also be established on a sprint-by-sprint basis, using commitment-based planning.
- Once established, velocity can be used to plan projects and forecast release and product completion dates.
Sprint Backlog

- Collect the agreed upon Sprint tasks in a Sprint Backlog
- Similar structure to the Product Backlog
- Team members sign up for tasks, they aren’t assigned by the Scrum Master
- Estimated work is entered in the Sprint Backlog
- Any team member can add, delete or change tasks on the Sprint Backlog during the Sprint
- Tasks for the Sprint emerge during the planning session and during the Sprint
- Tasks (content, estimates, sign-up) can change during the Sprint
An Sprint Burn down chart

![Sprint Burn down chart](image)

- **Completed Points**
- **Story Points Remained by the end of Iteration**

5.29.2015
Burn down chart continues...

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Mon</th>
<th>Tues</th>
<th>Web</th>
<th>Thu</th>
<th>Fri</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code the user interface</td>
<td>8</td>
<td>4</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code the middle tier</td>
<td>16</td>
<td>12</td>
<td>10</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Test the middle tier</td>
<td>8</td>
<td>16</td>
<td>16</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>Write online help</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TOTAL HOURS</strong></td>
<td>44</td>
<td>32</td>
<td>34</td>
<td>18</td>
<td>8</td>
</tr>
</tbody>
</table>

![Burn down chart](chart.png)
Agile-Scrum process at HelpSystems
Product Releases – Feature vs. Maintenance

- R&D Group of HelpSystems drives two kinds of product releases
  - Feature Release (With new feature and/or existing feature enhancements)
  - Maintenance Release (Product defect fixes)
Product Feature Release Process

1. Release Planning

2. User Story planning and estimation

3. Release kickoff

4. Agile-Scrum Process
   a) 2-weeks long Development and Testing Iterations
   b) Daily Stand up
   c) Iteration Wrap ups
   d) Product Demo during Iteration Wrap ups
   e) Iteration Retrospectives

5. Product Beta Release

6. Regression Testing
Product Maintenance Release Process

1. Release Planning

2. Detailed estimations on defects in the release scope

3. Agile-Scrum Process
   a) Daily Stand up

4. Product Beta Release

5. Regression Testing
Few last thoughts

• Agile transformation does not happen overnight
• Need to have
  – Goal for the transformation
  – Organization leadership buy in
  – Established minimal set of best practices
  – Management of expectation
References and Resources:

- Scrum
  - www.mountaingoatsoftware.com
  - www.controlchaos.com
  - www.scrumalliance.org
  - www.3back.com
  - www.collaborive
- Agile Software Development with Scrum
  - Ken Schwaber and Mike Beedle
- (GVU) Agile Project Management with Scrum
  - Ken Schwaber and Mike Beedle
- General information
  - www.agilealliance.com
  - www.agilemodeling.com
  - www.ambysoft.com
  - www.agiledata.org
  - www.enterpriseunifiedprocess.com
  - www.implementingscrum.com
Thank You