How to Utilize Agile Project Management for GIS Projects

Presented By
Lana Tylka and Jennifer Prather
How to Start…

1. Conduct kickoff meeting
2. Discuss similar industries
3. Assess workflows
4. Prioritize workflows
5. Choose a life cycle
6. Create a plan

Launching your Location Platform Guide: www.esri.com/LaunchGuide
Asses your experience…

- Does your organization have enterprise software implementation experience?
- Does your organization have GIS implementation experience?
- Does your organization have the necessary talent with knowledge to meet the roles and responsibilities?
Assess your current IT/GIS Capacity

- Hardware is cheap, experience is expensive
- Servers are easy, network is hard
- Beware of key components that you don’t control (NAS, firewalls, web accelerators…)
- Understand your vendor relationships
- Trust anecdotal information but verify
Asses your organization’s willingness to change

- Who will be impacted by Enterprise GIS changes?
- How will they be impacted and how will they respond?
- What mitigation options are available for negative impacts?
- Plan your strategies carefully & engage
- An Agile approach can be a massive transition
- An Agile approach can yield earlier successes
Identify roles, resources, and their availability

- What roles are needed to implement the Enterprise GIS?
- What would be their responsibilities?
- Identify in-house or contractor resources to potentially support the project as overseers or implementers
- Document roles, their responsibilities and candidates

GIS analyst
Program manager
System architect
Legal expert
Technical lead
Field collection specialist
Project manager(s)
Contract specialists
Account specialists
Database specialist
Subject matter experts
Business Analyst
SOE programmer
Formulate your capabilities and gaps

- List your assumptions
- List roles, responsibilities, potential resources and gaps
- List HW, SW and implementation gaps
- Identify mitigation options
Test and Plan for Growth

• Test your GIS for performance
  - So you know when it's out of compliance

• Monitor system metrics and availability
  - So you know about problems before your customers do

• Create 1/3/5 year plans
  - To ease staff augmentation, hardware and software acquisition

• Stay current on software versions
  - Operating System, ArcGIS Platform, Database instances, 3rd party applications

Known Gaps Can Drive Requirements
The Agile Approach
Agile Manifesto

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan
**Agile**: Agile development is an alternative to traditional project management where emphasis is placed on empowering people to collaborate and make team decisions in addition to continuous planning, continuous testing and continuous integration.

**User Story**: A user story is a very high-level definition of a requirement, containing just enough information so that the developers can produce a reasonable estimate of the effort to implement it.

**Sprint/Iteration**: In the Scrum method of Agile software development, work is confined to a regular, repeatable work cycle, known as a sprint or iteration. Scrum sprints used to be 30 days long, but today we advise one-week or two-week sprints.

**Epic**: An Epic can be defined as a work, which can not be completed in a week time, or any work which will take a full sprint to complete. By observation 5-10 user stories comprise of one Epic in agile methodology.
## When would you use Agile?

<table>
<thead>
<tr>
<th>Scope, Technology, Contract</th>
<th>Agile</th>
<th>Staged Delivery</th>
<th>Waterfall</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Flexible scope, deliverables</td>
<td>• Several applications</td>
<td>• Clear requirements</td>
<td></td>
</tr>
<tr>
<td>• One or several applications</td>
<td>• Prototypes expected</td>
<td>• Fixed deliverables</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Single application</td>
<td>• Single application</td>
<td></td>
</tr>
<tr>
<td>Size, Duration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Any size or duration project</td>
<td>• Medium or large size, mid to long duration</td>
<td>• Small size, short duration project</td>
<td></td>
</tr>
<tr>
<td>Capacity, Capabilities, Environment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Customer EXPECTS collaboration</td>
<td>• Capacity, resources, and environment to support multiple releases</td>
<td>• Limited capacity, resources, and environment</td>
<td></td>
</tr>
<tr>
<td>• Stable, experienced project team</td>
<td></td>
<td>• Frequent turnover on project team</td>
<td></td>
</tr>
</tbody>
</table>
Proposing Agile
Story point is an arbitrary measure used by Scrum teams. This is used to measure the effort required to implement a story. In simple terms, it's a number that tells the team how hard the story is. Hard could be related to complexity, unknowns, and effort. In most cases, a story point range is 1, 2, 3, 5, 8, 13, 21, 34, 45.
<table>
<thead>
<tr>
<th>Activity</th>
<th>Scrum Master</th>
<th>Product Owner</th>
<th>Developer</th>
<th>Analyst</th>
<th>System Admin</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Architecture</td>
<td>16</td>
<td>16</td>
<td>0</td>
<td>16</td>
<td>120</td>
<td>168</td>
</tr>
<tr>
<td>Geodatabase Design</td>
<td>24</td>
<td>24</td>
<td>0</td>
<td>184</td>
<td>40</td>
<td>272</td>
</tr>
<tr>
<td>Widget 1</td>
<td>24</td>
<td>24</td>
<td>176</td>
<td>48</td>
<td>0</td>
<td>272</td>
</tr>
<tr>
<td>Widget 2</td>
<td>20</td>
<td>20</td>
<td>240</td>
<td>80</td>
<td>0</td>
<td>360</td>
</tr>
<tr>
<td>Application Hardening</td>
<td>40</td>
<td>16</td>
<td>84</td>
<td>24</td>
<td>4</td>
<td>168</td>
</tr>
</tbody>
</table>
Work Breakdown Structure

Project

- System Architecture
  - 1.1 User Story
  - 1.2 User Story

- Database Design
  - 2.1 User Story
  - 2.2 User Story
  - 2.3 User Story

- Widget 1
  - 3.1 User Story
  - 3.2 User Story
  - 3.3 User Story
  - 3.4 User Story

- Widget 2
  - 4.1 User Story
  - 4.2 User Story
  - 4.3 User Story
  - 4.4 User Story
  - 4.5 User Story

- Application Hardening
  - 5.1 User Story
  - 5.2 User Story
Managing Agile
KanBan Approach (Still Agile, just not Scrum)

- No defined iterations
- No defined roles
- Direct communication with customer
- Limit your work-in-progress
- Visualize your work
- Ever-changing backlog with on-the-fly prioritization
As a [role], I can [feature] so that [benefit]
As a field representative, I want to collect information offline so that data can be collected in remote locations.
### Product Backlog

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Sprint Backlog
Using Agile in a Professional Services Project
Using Agile in a Consulting Project

Method

Waterfall

Agile

Time
Using Agile in a Consulting Project
Using Agile in a Consulting Project
Using Agile in a Consulting Project
Using Agile in a Consulting Project

Method

Waterfall

Agile

Time

Final Release
Managing Resources

Plan A

Plan B

Plan Z

Your Project

Sprint

100% 50%
50% 75%
75% 100%
100% 75%
50% 50%
50% 75%
75% 100%
100% 75%
50% 50%
50% 75%
75% 100%
100% 75%
Keys to Successful Projects!

Communication

Trusted Partnerships

Transparency

Utilize Available Tools…
Tools
Using Trello
Using GitHub
Using TFS

Backlog items to Features

<table>
<thead>
<tr>
<th>Title</th>
<th>State</th>
<th>Effort</th>
<th>Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquisition and core workflow</td>
<td>In Progress</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Add, Edit, Remove from shopping cart</td>
<td>Approved</td>
<td>5</td>
<td>Mobile</td>
</tr>
<tr>
<td>Edit existing users</td>
<td>Committed</td>
<td>5</td>
<td>Web</td>
</tr>
<tr>
<td>Direct link to an item in the catalog</td>
<td>Committed</td>
<td>8</td>
<td>Mobile</td>
</tr>
<tr>
<td>Welcome mails that include a “Getting Started” v...</td>
<td>New</td>
<td>3</td>
<td>Mobile</td>
</tr>
</tbody>
</table>

Tasks Needing Attention

- To Do
- In Progress

Backlog Distribution

<table>
<thead>
<tr>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Committed</td>
</tr>
<tr>
<td>Approved</td>
</tr>
<tr>
<td>New</td>
</tr>
<tr>
<td>Done</td>
</tr>
</tbody>
</table>

Title

- Editing items already added to the cart
- Cancelling a submitted order

<table>
<thead>
<tr>
<th>Title</th>
<th>State</th>
<th>Remaining Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notification to the user</td>
<td>To Do</td>
<td>6</td>
</tr>
<tr>
<td>Design work on the cancel dialog</td>
<td>To Do</td>
<td>4</td>
</tr>
<tr>
<td>Credit card refund</td>
<td>To Do</td>
<td>4</td>
</tr>
<tr>
<td>Order history updated styling</td>
<td>To Do</td>
<td>4</td>
</tr>
<tr>
<td>End-to-end testing</td>
<td>To Do</td>
<td>2</td>
</tr>
<tr>
<td>New route to view a cancelled order</td>
<td>To Do</td>
<td>2</td>
</tr>
</tbody>
</table>

Work By: Assigned To

- Christina Kelly (8 of 14 h)
- Lowell Steel (17 of 14 h)
- Noah Munger (6 of 7 h)

Type of query

- Flat list of work items
- Work items and direct links
- Tree of work items

<table>
<thead>
<tr>
<th>Field</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team Project</td>
<td></td>
<td>©Project</td>
</tr>
<tr>
<td>Assigned To</td>
<td></td>
<td>Christina Kelly</td>
</tr>
<tr>
<td>Work Item Type</td>
<td></td>
<td>Task</td>
</tr>
<tr>
<td>Backlog Priority</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blocked</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Business Value</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Save query

- Called By
## Making a Decision

<table>
<thead>
<tr>
<th>Project Considerations</th>
<th>Trello</th>
<th>GitHub</th>
<th>TFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements are Proprietary</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Mobile App</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Easy to setup</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Estimation tools</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Scheduling tools</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Automated Burndown chart</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Easily integrated with Visual Studio for Code Repository</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Capacity Planning</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Exports to MPP and Excel</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
</tbody>
</table>
Case Studies
Small Scale

<table>
<thead>
<tr>
<th>Contract Type</th>
<th>$$ Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>T&amp;M</td>
<td>$110K</td>
</tr>
</tbody>
</table>

- **Product Owner**
- **Customer’s PM**
- **Scrum Master**
- **Lead developer**
- **The team**
  - Dev team of 2 UI/UX as needed
  - Project Manager
- **Stakeholders**
  - Customer’s PM
- **Product Backlog**
- **Sprint Planning**
- **Sprint Backlog**
- **Retrospective**
- **Potentially Shippable Product Increment**
- **Daily Scrum**
- **3 Day Sprint**
Case Study – Small Scale

• Why Agile?
  - Requirements (User Stories) are not clearly defined at the time of contract award.

• Key Challenges / Lessons Learned
  - Stakeholders (customer) was an active participant with respects to the grooming of the product backlog including prioritization.
  - Standard sprints do not work with the customer’s schedule as the work comes in waves rather than a steady pace.
  - Finding resources to staff a project like this can be difficult since the work is not planned out well in advance.
Case Study – Medium Scale

<table>
<thead>
<tr>
<th>Contract Type</th>
<th>$ Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>T&amp;M</td>
<td>$4M</td>
</tr>
</tbody>
</table>

**Product Backlog**

**Sprint Planning**

**Sprint Backlog**

**Daily Scrum**

**1 Week Sprint**

**Potentially Shippable Product Increment**

**The team**

- 12 developers
- 2 testers
- 1 PM
- Fluctuates as needed

**Stakeholders**

- Product Owner
- Scrum Master
- Lead developer
- Analyst

**Product Owner**

**Scrum Master**

**Lead developer**

**Analyst**

**The team**

**Customer’s PM**

**Internal PM**

**Retrospective**

**Case Study – Medium Scale**

**Contract Type**

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>T&amp;M</td>
<td>$4M</td>
</tr>
</tbody>
</table>
Case Study – Medium Scale

• **Why Agile?**
  - Customer was familiar with Agile and believed iterations was the best method to get to realize their end goal

• **Key Challenges / Lessons Learned**
  - Stakeholders (customer) was an active participant with respects to the grooming of the product backlog and sprint planning events.
  - Team consisted of contractors from multiple companies who were all using their own version of Scrum
  - Utilization of multiple contractors created dependencies that had to be accounted for in Sprint Planning.
  - Hours were used for estimates to avoid an inconsistent Points experience
  - Monthly iterations, then bi-weekly, then weekly, then back to bi-weekly in order to get the right amount of feedback
Case Study - Large Scale

<table>
<thead>
<tr>
<th>Contract Type</th>
<th>$ Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFP-LOE</td>
<td>$9M</td>
</tr>
</tbody>
</table>

### Stakeholders
- Customer’s PM
- Project Manager
- Analyst
- Lead developer
- 7 developers
- 1 tester

### Processes
- Daily Scrum
- Sprint Planning
- Retrospective
- Scrum of Scrums
- Potentially Shippable Product Increment
- Product Backlog
- Scrum of Scrum

### Visualization
- The team consists of analysts, developers, and product owners.
- Development teams are led by a release manager and a scrum master.
- "~7 developers 1 tester" suggests a team structure.
- Contract type: FFP-LOE, valued at $9M.
Case Study – Large Scale

• **Why Agile?**
  - Project was contractually required to follow the SAFe Agile Methodology.
  - Requirements were vague and customer recognized the benefit in iterative development to achieve the best results.

• **Key Challenges / Lessons Learned**
  - Deployment into the customer’s footprint occurs at the end of the Release.
  - Large project team to manage.
  - Each Scrum Team was responsible for individual features.
  - Dependencies existed between scrum teams.
  - Stakeholders (customers) were only present during Stakeholder Reviews and were not active participants during the release planning events.
  -Disconnected environment meant that the customer could not test the features until the end of a release.
  - Bi-weekly demonstrations to “sell off” features and to show progress.
Questions?
Please Take Our Survey on the **Esri Events App**!

Download the Esri Events app and find your event

Select the session you attended

Scroll down to find the survey

Complete Answers and Select “Submit”