How to Utilize Agile Project Management for GIS Projects

Lana Tylka and Jennifer Prather
How to Start…

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

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

**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.

**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.
## 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>• Single application</td>
<td></td>
<td>• Single application</td>
<td></td>
</tr>
<tr>
<td>Size, Duration</td>
<td>• Any size or duration project</td>
<td>• Medium or large size, mid to long duration</td>
<td>• Small size, short duration project</td>
</tr>
<tr>
<td>Capacity, Capabilities, Environment</td>
<td>• Customer EXPECTS collaboration</td>
<td>• Capacity, resources, and environment to support multiple releases</td>
<td>• Limited capacity, resources, and environment</td>
</tr>
<tr>
<td></td>
<td>• Stable, experienced project team</td>
<td></td>
<td>• Frequent turnover on project team</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.
## Pricing Sheet

<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
Scrum Sprint Cycle

Product Backlog

Sprint Planning

Sprint Backlog

The team

Scrum Master

The team

Product Owner

Stakeholders

Daily Scrum

Potentially Shippable Product Increment

Retrospective

2 - 4 Week Sprint
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.
<table>
<thead>
<tr>
<th>Wish List</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
<tr>
<td>✔️</td>
</tr>
</tbody>
</table>
Sprint Backlog

- 4h
- 8h
- 3 days
- 2h
- 1h
- 2 days
- 4h
- 8h
Using Agile in a Professional Services Project
Using Agile in a Consulting Project
Using Agile in a Consulting Project

- Waterfall
- Agile

Time
Using Agile in a Consulting Project

- Waterfall
- Agile

Time
Using Agile in a Consulting Project

Method
- Waterfall
- Agile

Time
Using Agile in a Consulting Project

Method

Waterfall

Agile

Time

Sprint 0.0
Sprint 0.1
Sprint 0.2
Sprint 1.1
Sprint 1.2
Sprint 1.3
Sprint 2.1
Sprint 2.2
Sprint 2.3
Sprint 3.1
Sprint 3.2
Sprint 3.3
Sprint 4.1
Sprint 4.2
Sprint 4.3
Final Release
Managing Resources

Plan A

Your Project

Plan B

Sprint

100% 50%
50% 50%
75% 100%
100% 75%

Plan Z

Sprint

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

Communication

Utilize Available Tools…

Trusted Partnerships

Transparency

Utilize Available Tools…
Tools
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>New</td>
<td>Mobile</td>
</tr>
<tr>
<td>Welcome mails that include a &quot;Getting Started&quot; v...</td>
<td>New</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

### Title

<table>
<thead>
<tr>
<th>Title</th>
<th>State</th>
<th>Remaining Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Editing items already added to the cart</td>
<td>Done</td>
<td>9</td>
</tr>
<tr>
<td>Cancelling a submitted order</td>
<td>Committed</td>
<td>22</td>
</tr>
<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>

### Tasks Needing Attention

- To Do
- In Progress

### Backlog Distribution

- Committed: 29
- Approved: 11
- New: 5
- Done: 12

### Type of query

- And/Or
- Field
- Operator
- Value
- Team Project
- Assigned To
- Work Item Type
- Backlog Priority
- Blocked
- Business Value
- 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>
Small Scale

**Contract Type** | **$$ Value**
--- | ---
T&M | $110K

- **Product Owner**
- **Customer’s PM**
- **Scrum Master**
- **Lead developer**
- **Dev team of 2 UI/UX as needed Project Manager**
- **The team**
- **Stakeholders**
- **Customer’s PM**

**Product Backlog**

**Sprint Planning**

**Sprint Backlog**

**Potentially Shippable Product Increment**

**Daily Scrum**

**3 Day Sprint**

**Retrospective**
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**
- **Product Owner**
- **Analyst**
- **Lead developer**
- **Scrum Master**
- **The team**
  - 12 developers
  - 2 testers
  - 1 PM
  - Fluctuates as needed
- **Sprint Planning**
- **Sprint Backlog**
- **Daily Scrum**
- **1 Week Sprint**
- **Potentially Shippable Product Increment**
- **Retrospective**
- **Stakeholders**
  - Customer’s PM
  - Internal PM
- **Visual Studio**

**Contract Type**
- T&M
- $4M
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>

- **Product Backlog**

- **Release Manager**
  - Analyst
  - Lead developer

- **Scrum of Scrum Master**
  - The team
    - ~7 developers
    - 1 tester

- **Sprint Planning**
  - Sprint Backlog
  - Daily Scrum
  - 2 Week Sprint

- **Retrospective**
  - Potentially Shippable Product Increment

- **Stakeholders**
  - Customer’s PM
  - Project Manager

- **Analyst**
  - Developer

- **Functional Requirements**

- **Visual Studio**

- **Master**

- **Value**

- **Target Audience**

- **Project Timeline**

- **Acceptance Criteria**

- **Technical Specifications**

- **Legal Considerations**

- **Risk Management**

- **Quality Assurance**

- **Scrum of Scrums**
  - Analyst
  - Developer
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 App

Download the Esri Events app and find your event

Select the session you attended

Select the Feedback tab

Complete answers and select “Submit”