

Goal

Goal: Popularize the “user story” approach to support scientific software development (in particular for DOE’s Exascale Computing Project, <https://www.exascaleproject.org>)

Benefit: A user story is a flexible, simplified description of a requirement of a product or project that can assist in planning and discussion. It accommodates the evolution of requirements and solutions in collaborative efforts towards a finished product.

Basic Elements:

- ❑ **Who** is the user?
- ❑ **What** is required?
- ❑ **Why** is the requirement needed?

short sentences
(Agile methodology)

Strategy: The IDEAS-ECP project has been pursuing the user story approach to streamline the project’s activities (for prioritizing and planning work) and also to understand how user stories could support software development efforts by ECP teams.

Workflow: User stories in the development of methodologies material (example).



PSIP: Productivity and Sustainability Improvement Planning

IDEAS-ECP User Story Categories

The user stories used in IDEAS-ECP are classified as follows:

- ❖ **training & documentation**
 - casual usage of services for version control
 - design patterns for version control
 - new features in a programming language
- ❖ **software integration & testing**
 - component integration and testing
 - software robustness
 - multi-repository development
- ❖ **software quality**
 - updates on HPC architectures
 - continuous integration to enable testing at DOE computing facilities
 - access to reliable tools for improving application performance
- ❖ **practice & standards**
 - creation of team policies
 - coding standards
 - guidelines on releasing, licensing, copyrighting
 - management of software contributions
- ❖ **software requirements & development**
 - time-saving tips for developing better software
 - improvement of design processes
 - code reviews for producing higher quality software
- ❖ **operational**
 - editorial workflow for Better Scientific Software site (<https://bssw.io>)
 - interviews performed by the project
 - documenting and publicizing the project’s outreach activities

IDEAS-ECP User Story Sample

category	example
training and documentation	<i>As a casual user of GitHub, I want more GitHub tutorials and tips so that it becomes easier for me to recall the functionality.</i>
software integration and testing	<i>As an application architect, I want to better understand version control capabilities that allow integration of independently developed components so that we can distribute a coherent software stack.</i>
software quality	<i>As an ECP developer, I want documentation and training in setting up automated testing for my package as well as using my package testing within the ECP CI system so that I can reliably determine and regularly track that various of my package branches compile and minimal tests pass in all configurations and machines relevant to other ECP users</i>
practices and standards	<i>As a person responsible for software quality and correctness for my project, I want guidance on selecting and implementing coding standards so that we can make our code easy for everyone to read and understand</i>
software requirement and development	<i>As a software engineer in HPC, I want to connect test development to software design so that intertwined dependencies that get in the way of building stand-alone tests can be minimized</i>
operational	<i>As a participant in the CSE software engineering community, I want a documented process for contributing to the bssw.io website so that I can add my knowledge to the site in an efficient way</i>

User Story Evolution

- Story spans the categories of software quality and software integration and testing
- Story began from the perspective of an ECP developer and unfolded into five tickets for resource development
 1. creation of an example package demonstrating the basics of automated testing
 2. extension of the example package to support different running configurations
 3. extension of the example package to handle variation in test results
 4. extension of the example package to support different testing systems
 5. extension of the example package to support performance analysis

Impact and Future Directions

- ❖ Adoption by ECP Software Technology Focus Area (sample)
 - *As a developer responsible of QA, I want to be able to setup continuous integration testing on LCF systems (production or testing)*
 - *As a software project lead, I would like a way to be informed of ECP application needs from ST projects*
 - *As an xSDK customer, I want a documented process describing how to submit feature requests so that I know that my requests are fully considered.*
 - *As an xSDK customer, I want preinstalled xSDK software on LCF machines so that I can easily compile/run code on them.*
- ❖ Additional information: <https://bssw.io/items/user-stories>

xSDK: Extreme-scale Scientific Software Development Kit, <https://xsdk.info/ecp>

References

- Sletholt, Hannay, Pfahl and Langtangen, *What Do We Know about Scientific Software Development's Agile Practices?* Computing in Science Engineering, 14:24-37, 2012.
- Heaton and Carver, *Claims about the use of software engineering practices in science: A systematic literature review*, Information and Software Technology, 67:207-219, 2015.
- <https://www.mountangoatsoftware.com/agile/user-stories>
- How to Write Good User Stories (CA Technologies): <https://youtu.be/tKSUokG3Y0w>
- Kniberg and Skarin, *Kanban and Scrum - Making the Most of Both*, InfoQ, 2010.
- Interoperable Design of Extreme-scale Application Software (IDEAS): <https://ideas-productivity.org>