

# A Business Analyst's Experience With Scrum

I recently wrote a book detailing a customizable extension to the Scrum framework. The following diagram provides an overview of the process described in that book.

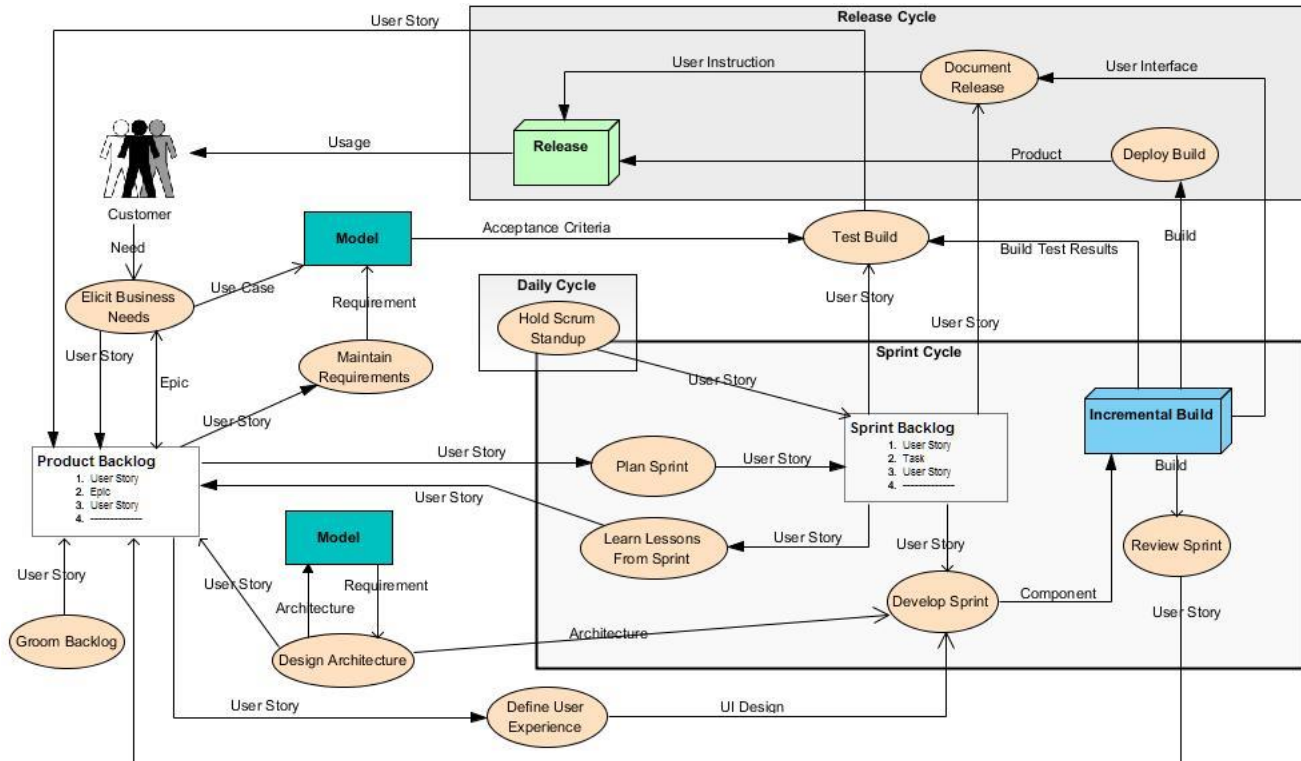


Figure 1: Extension To The Scrum Framework

This article provides an overview of the process and the reasons why I created it. If you find this information useful and want to learn more, I provide a reference to my book at the end.

## 1 Introduction

Working as a business analyst on *agile* projects for 6 years, I have learned that the Scrum process framework does not describe the complete story.

---

**I emphasized the word *agile*, to indicate that the one thing all of these projects have in common are short time-boxed development cycles (or sprints).**

---

Scrum identifies three roles. The development team, product owner and Scrum Master. Scrum does not identify roles for the business analyst, solution architect, quality assurance team, tester, UI designer, deployment manager or writer. (These are roles that I am used interacting with on an agile project.) Instead, with Scrum, the development team is responsible for the work normally performed by these roles. Scrum assumes that the development team includes people with all of the skills normally associated with these roles.

---

**Quoted from Scrum.org: Development Teams are cross-functional, with all the skills as a team necessary to create a product Increment. Scrum recognizes no titles for Development Team members, regardless of the work being performed by the person. Scrum recognizes no sub-teams in the Development Team, regardless of domains that need to be addressed like testing, architecture, operations or business analysis. Individual Development Team members may have specialized skills and areas of focus, but accountability belongs to the Development Team as a whole.**

---

A problem with this Scrum organization is that all of the work performed by the development team happens within a sprint cycle. However, the artifacts produced by the product owner, business analyst, solution architect, quality assurance, writer and deployment manager, are not sprint based. In addition, the UI designer and tester activities are sometimes independent of a sprint.

This article describes activities performed outside of the sprint cycle and identifies the benefits that they bring to the implementation of a deliverable product.

## My Experience With Agile Projects

In my experience, some agile projects included a Scrum Master and other times project team members perform the Scrum Master activities. The roles that all of these projects had in common are, a product owner (sometimes known as the product manager), a development team, and of course, a business analyst.

Most development teams were supported by a separate user interface design team and an independent quality assurance (QA) or testing team. Other roles that are outside of development include deployment and operations, solution architects and document writers. These roles may be assigned specific responsibilities that support software development, even when an agile process is employed.

This article contains an overview of the activities, deliverables and best practices for these roles as they relate to agile systems development. Each role is responsible for one or more activities. These activities are described from a business analyst's point of view.

## A Brief Overview Of The Scrum Framework Process

Scrum identifies five activities (ceremonies), Hold Scrum Standup, Plan Sprint, Develop Sprint, Review Sprint and Learn Lessons From Sprint (Sprint Retrospective). The artifacts identified by Scrum are, user stories, software components and an incremental build. User stories are managed with the product backlog and the sprint backlog. All of this work is performed within two time-boxed cycles (Daily and Sprint).

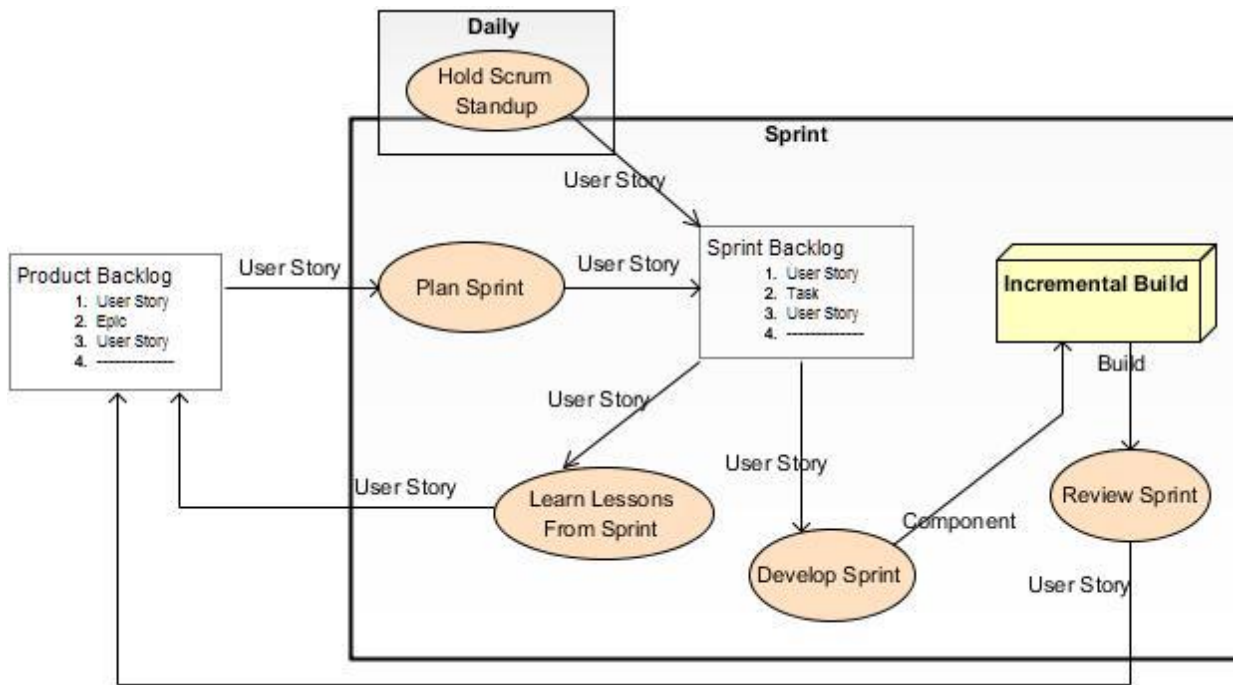


Figure 2: Scrum Framework

User stories are entered into a product backlog. The product backlog is managed by the product owner. User stories are taken into the next sprint cycle during a sprint planning meeting. During the sprint cycle, the development team develops components for an incremental build. When all user stories in the sprint backlog are done, the incremental build is reviewed with project stakeholders. After the sprint is complete, a sprint retrospective meeting is conducted to identify lessons learned from the previous sprint. A standup meeting is held every day, for the development team to plan the next day's work and review the status of user stories in the sprint backlog.

User stories may be generated as a result of the sprint review or sprint retrospective and entered into the product backlog.

Development of sprint backlog user stories includes, building user and system interfaces, updating the system architecture, writing user documentation, testing software components, testing the incremental build and deploying software to the customer.

The Scrum framework does not show activities that are the responsibility of the product owner, such as eliciting business needs as user stories and grooming the backlog, in order to prioritize work for upcoming sprints. These activities occur outside of the development sprint cycle.

Epics are user stories that are too large for a single sprint. Tasks are the breakdown of user stories into manageable pieces of work.

## 2 My Agile Experience As A Business Analyst

The process that is described below includes artifacts, roles, and activities that I have been involved with when working as a business analyst on 'agile' projects.

As a business analyst, I generally support the product owner by:

- Eliciting business needs
- Maintaining the user stories in the product backlog
- Writing and maintaining project documentation
- Modeling business processes and system requirements, architecture and data

In addition, the business analyst may be expected to support development by:

- Assisting with testing, by writing acceptance criteria and confirming test case results
- Attending Scrum ceremonies, including the daily standup
- Supporting development as requested

As you can see, the BA may be expected to provide support to all activities in an agile development process. (I have even been asked to sign-off incremental builds prior to delivery to a customer.)

## Team Logistics

A major factor that affects the Scrum process is the logistics (or organization) of the project team and its stakeholders. I have worked in environments with the following logistical makeup:

- The customer is remote – When the development team does not have direct access to the customer, the product owner (or similar role) acts as a proxy for the customer. The product owner may be the only person on the team who has direct contact with the customer. In this case, they are fully responsible for identifying customer needs and priorities and for confirming the accuracy of the product details. The customer may only get to see the product when it is released.
- The development team is remote - Many projects do not have access to a co-located IT department. Software development may be subcontracted to a development team at a remote location. If the development team is located in a different country, then direct communication may be restricted to a few hours a week.

---

**I have worked with a development team whose working hours were offset from the customer by 36 hours per week. By the time I arrived at work on a Monday morning, the developers had been working for a day and a half.**

---

- Quality assurance is a separate department – The project does not have dedicated testers. Testers are part of a QA department that is a shared resource across all departments. The development team will test each incremental build, but no software can be released to a customer until QA has validated that the product meets the customer needs. Product acceptance testing occurs after the sprint is complete and prior to product release.
- A team of UX experts designs user interfaces – To ensure that customers experience a common look and feel across the enterprise, all user interface design work is performed by an independent UX department.

Product UI specifications (or mockups) are included with the user stories that are taken into sprint planning. The UI specifications may change during development, with the approval of the UI designers.

- Product deployment is on a separate release cycle - The customer does not want any disruption to their business due to defects or problems with deployment. Therefore, deployment occurs at the customer's convenience and only after user acceptance testing is done.
- Product documentation is part of the product release – User manuals and release notes are only written against a release. Documentation is not necessary for an incremental build, and therefore it does not need to occur as part of a sprint.

Assuming the worst case, where all of these traits are affecting your development process, I documented the following extensions to the Scrum process to show how they affected my experience with Agile.

### 3 Overview Of The Process

The process diagram assumes that all the above logistical items are affecting the development process. The process is customized according to your development environment. For example, if UI design is performed during a sprint as part of development of a user story, then the Define User Experience activity may be removed from the process.

#### The Activities

In the following paragraphs, I describe how to read the process diagram shown in figure 1.

There are 3 time-boxed cycles shown in the process; Daily Cycle, Sprint Cycle, Release Cycle, and a 4<sup>th</sup> region where activities are performed on a continuous basis. The activities shown in these regions are connected by flows that indicate artifacts passed between them. The flow of information is continuous and it is passed in increments. (The diagram does not indicate any hand-offs from one role or department to another.) Four repositories of information are shown. Information may be added to, updated and extracted from these repositories at any time.

Continuous activities:

- Elicit Business Needs - Business needs are elicited from the customer and captured in the product backlog as epics. Epics are analyzed to derive use cases and user stories. User stories are managed within the product backlog. Use cases are maintained within the project model.
- Groom Backlog - The product backlog is reviewed on a regular basis and user stories are prioritized. Backlog items may be approved for development, or removed from the backlog.
- Maintain Requirements - Requirements are continuously maintained within the Model. Use cases are analyzed, user stories updated in the product backlog and acceptance criteria are derived.
- Design Architecture - The system architecture is maintained as new technology is introduced and interfaces are changed. A model architecture view captures the system architecture. User stories are maintained in the product backlog to capture architectural changes. Model architecture diagrams may be used during development of sprint user stories.
- Define User Experience - System user interfaces are updated from product backlog user stories. Mockups of the user interface design are taken into the sprint and may be updated as developers work on the user stories.
- Test Build - Test cases are continuously developed from user story acceptance criteria, so that an incremental build may be tested as soon as it is available. Test results are captured as a result of testing an incremental build.

Sprint cycle activities:

- Plan Sprint - User stories are taken into the next sprint backlog during sprint planning, according to their priority in the product backlog.
- Develop Sprint - Each user story impacts one or more components of the product build. During a sprint cycle, all stories in the spring backlog are developed and an incremental build is produced.

- Review Sprint - When an incremental build has been produced, it is reviewed with the project stakeholders. User stories may be updated in the product backlog as a result of demonstrating the results from the sprint.
- Learn Lessons From Sprint - The final activity in a sprint is to hold a lessons learned meeting with all members of the project team. The results of the sprint are input to the meeting and action items may be created as user stories in the product backlog.

Release cycle activities:

- Document Release - Each release includes instructions for the users and release notes. These documents, inform the customer what has changed since the previous release. Release notes and updates to user manuals are derived from a combination of the developed user stories that make up the release and from exposure to the software user interfaces.
- Deploy Build - The product is delivered to the customer on an agreed schedule, so that deployment does not interfere with their business activities. Deployment may involve, stopping running software, installing to the customer environment, interfacing with existing systems, validating that the installation was successful and restarting the system.

Daily cycle activities:

- Hold Scrum Standup – Members of the project team present their work since the last standup, their plans for the coming day's work and impediments to progress. User stories statuses are updated in the sprint backlog.

## The Artifacts

Artifacts are the things that are produced within the process. Artifacts may be maintained within a project repository, such as the Model, Product Backlog or Sprint Backlog. Other artifacts may be temporary and used to distribute information between team members. Artifacts contribute towards producing a product release.

---

**Not shown in the diagram, are repositories for documentation, code, and test cases, amongst others.**

---

The process diagram shows the following artifacts as input to or output from one or more activities:

- Needs - Elicited from the customer, business needs are analyzed into user stories for the product backlog.
- Epic – Business needs are captured in the product backlog as epic type user stories. Epics are broken into user stories. They are not pulled into a development sprint.
- User Story – All work performed in a sprint is the result of a user story. User stories result from; the analysis of epics, defects found during testing or may be created by the project team to capture work that needs to be done.
- Use Case – Captures the business processes that are derived from business needs. They are maintained in the project model. Use cases are analyzed to produce product requirements, data and user stories.
- Requirement – This is a generic term for behavior the system must exhibit. Requirements are documented with use cases, acceptance criteria, user stories and any other means the team uses to capture requirements.
- Architecture – The combination of software, hardware and their interfaces that will host an incremental build. The system architecture is captured in the project model and used by developers during the sprint in order to understand the impacts software changes. The solution architect works with the development team to identify impacts to sprint user stories.
- UI Design – Normally in the form of user interface mockups, the user interface design is attached to user stories in the product backlog. The UI Designer works with the development team during the sprint.
- Acceptance Criteria - Derived from use cases, they detail behavior that the product will exhibit in order to meet its requirements. Acceptance criteria are used to create and maintain product test cases.
- Component - A piece of software that is created or updated from a user story. Components are added to an incremental build during a sprint.
- Build - An instance of the software application that is updated every sprint.
- Build Test Results - Captured in the product test cases as the results of applying the test case to a product build. Each incremental build is tested against the user stories that were developed during the sprint.

- Product - A validated software build that and can be deployed to a customer.
- Usage - Refers to the customer experience with a product release.
- User Instructions – Documentation is written for anyone who needs access to the deployed product.

## The People

A role encapsulates a set of responsibilities performed by one or more people. A person may perform several roles. The roles that were previously identified, are the business analyst, product owner, solution architect, UI designer, development team, tester, writer and deployment manager. Each role is responsible for one or more activities in the process.

---

**The Scrum Master is considered out of scope. Responsibilities of the Scrum Master as they relate to the process, are distributed across other roles on the project team.**

---

The business analyst is responsible for the Elicit Business Needs and Maintain Requirements activities. They ensure that user stories are ready for the next sprint. They update user stories from the system architecture and UI design and provide support to development, testing and anyone using requirements. The business analyst works closely with the product owner to ensure the success of the project. The business analyst creates and maintains a model of the Requirements and the model Architecture view. They create User Stories, Acceptance Criteria.

The deployment manager is responsible for the Deploy Build activity. They support customer and product deployment at the customer location. The deployment manager creates the Product Release.

The development team ensures the success of each sprint. They are responsible for the Develop Sprint, Plan Sprint, Review Sprint, Learn Lessons From Sprint and Hold Scrum Standup activities. The development team creates Components for an Incremental Build and may create work items in the form of User Stories, for the product backlog.

The product owner is responsible for the Groom Backlog activity. They prioritize and ensure the accuracy of business needs and serve as a proxy for the customer and subject matter experts, and remove obstructions to development. The product owner creates Epics.

The solution architect is responsible for the Design Architecture activity. They maintain the hardware and software architecture of the complete solution. The solution architect creates the product Architecture, which is captured in the model Architecture view.

Quality assurance is a separate department within the organization that includes testers. They are responsible for the Test Build activity. QA ensures the quality of every artifact that is produced during the process including validation of all software builds. QA creates test cases from acceptance criteria and user stories. During validation of an incremental build, they populate test cases with Build Test Results.

The UI designer is a member of an independent User Experience (UX) design team. They are responsible for the Design User Interface activity. The UI designer produces user interface mockups (UI Design) that provide a consistent and efficient look and feel to the product user interfaces.

The writer documents clear, understandable and accurate user documentation that supports a product release. The writer is responsible for the Document Release activity. They create User Instructions for a product release.

## 4 Summary

This article provided a summary of a process that extends Scrum by introducing roles that are responsible for business analysis, solution architecture, UI design, quality assurance, documentation, and deployment. It defines the artifacts that are the responsibilities of these roles and the activities used to generate these artifacts.

These activities are only necessary when the role exists in the project team. Where a role is not applicable to your situation, that role's work may be performed by the development team instead.

Even if your project does not explicitly identify the artifacts described in this process, the development team probably performs the work described by the activity that produces the artifact.

The process may be customized for projects using Kanban, SAFe or LEAN project. Details are documented in my book, *Using Agile In A Quality Driven Environment (A Business analysts Experience With Scrum)*.

For more information, you may download or purchase my book from Amazon or Lulu.com. It is available for Amazon [Kindle](#), in [paperback](#), or as a free downloadable [PDF](#) file.

More information is on my website <http://www.lmunday.com>.