

Release Management Playbook

A hand-drawn diagram in white chalk on a grey background. It features a central rectangular box containing the title 'Release Management Playbook'. Surrounding the box are several 'X' marks and a series of circles connected by arrows, forming a circular flow that represents a process or lifecycle.

Table of Contents

● Software Delivery Framework	3	● Ceremonies	15
● Roles	4	○ Ceremony Cadence	16
○ Business Sponsor	5	○ Intake	17
○ Business Partner	6	○ Feature Review	18
○ Business Relationship Manager	7	○ Deployment Planning	19
○ Release Manager	8	○ Deployment Scrum	20
○ Program Manager	9	○ Defect Triage	21
○ Deployment Manager	10	○ Change Advisory Board	22
○ Development Manager	11	○ Deployment	23
○ TQA	12	○ Deployment Retro	24
○ UAT / DIT	13	● Appendix	25
○ Product Owner	14	○ Feature Template	26
		○ User Story Template	27
		○ Glossary	28

Software Delivery Framework

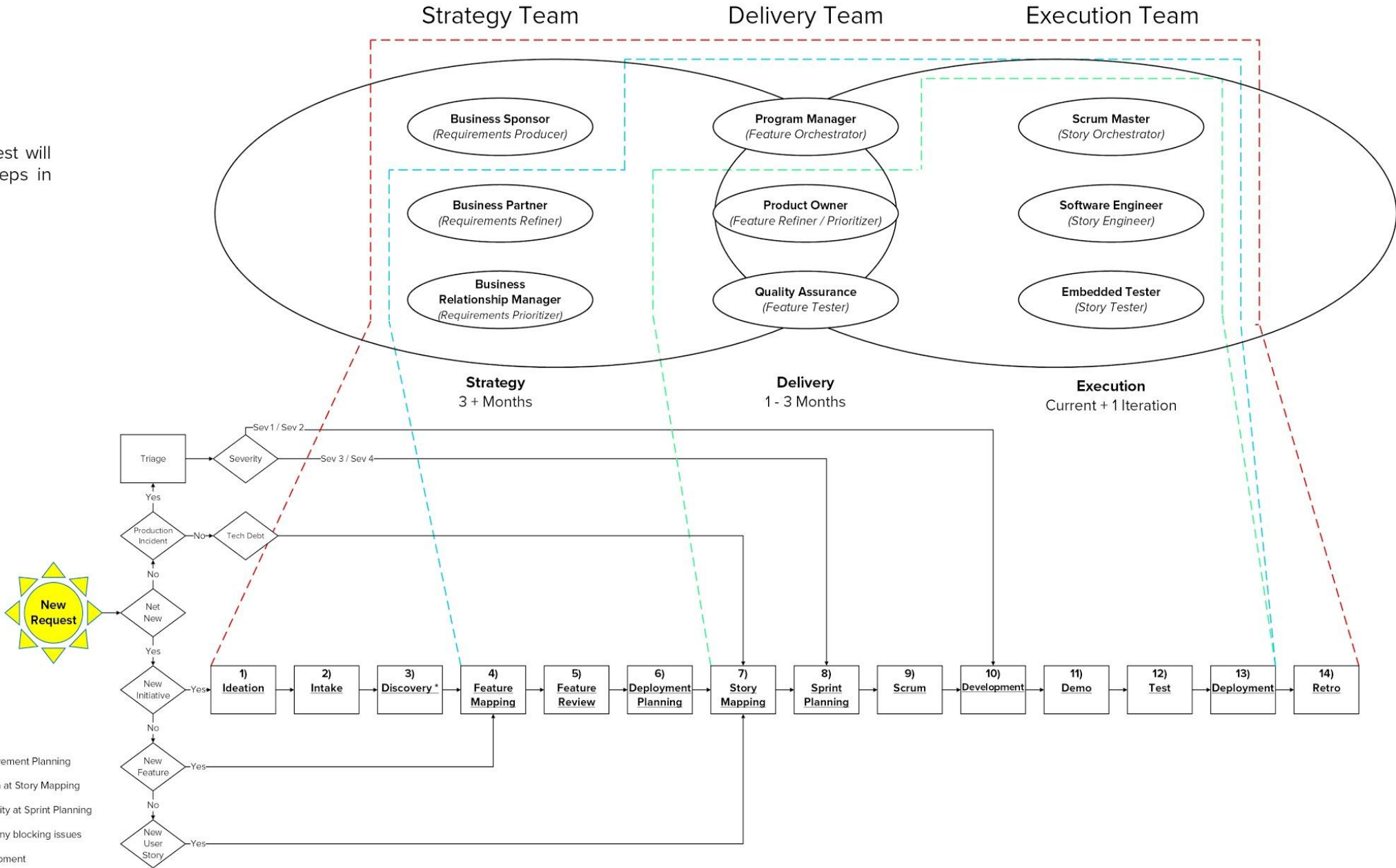
The type, size, and complexity of a request will determine who is impacted and what steps in the process need to be followed

Legend

- **New Initiative**
 - Impact 1+ Product
 - Impact 1+ Team
 - Deployed in 1+ Milestone
- **Existing Initiative**
 - New Feature**
 - Impact 1+ Team
 - Deployed in 1 Milestone
 - >300 Hours of Development
 - Existing Feature**
 - Impact 1 Team
 - Deployed in 1 Milestone
 - >20 Hours of Development
- **Existing Initiative**
 - Existing Feature**
 - Impact 1 Team
 - Deployed in 1 Milestone
 - >20 Hours of Development
 - New Story**
 - Impact 1 Team
 - Deployed in 1 Milestone
 - >20 Hours of Development

Process Steps by Project Phase

- 1) New Projects created in Ideation
- 2) Impacted Teams identified at Intake
- 3) Requirements identified in Discovery
- ● 4) Features defined in Feature Mapping
- ● 5) Feature Scope accepted in Feature Review
- ● 6) Incremental Feature delivery priority finalized at Production Increment Planning
- ● 7) Required Stories identified to meet Feature Acceptance Criteria at Story Mapping
- ● ● 8) Execution Team commits to top priority Stories based on capacity at Sprint Planning
- ● ● 9) Review of what has been completed, what is in progress, and any blocking issues
- ● ● 10) Execution Teams completes committed Stories during Development
- ● 11) Features are presented for approval at Demo
- ● 12) Features pass TQA / UAT / DIT testing during Test
- ● 13) Code Deployed to Production
- 14) Teams utilize Retros for continuous improvement



Roles

- **Business Sponsor (BS)** / Requirements Producer / Approver
- **Business Partner (BP)** / Requirements Refiner
- **Business Relationship Manager (BRM)** / Requirements Prioritizer
- **Release Manager (RM)** / Deployment Orchestrator
- **Program Manager (PgM)** / Feature Orchestrator
- **Deployment Manager** / Deployment Driver
- **Development Manager** / Team Leader
- **TQA** / Technical Testing
- **UAT & DIT** / Functional Testing
- **Product Owner (PO)** / Feature Driver

Business Sponsor / Requirements Producer

Responsible For:

- Establishing Portfolio Roadmap
- Creating Demand
- Identifying Business Case
- Obtaining VP Sponsor Approval for Demand
- Defining Project Value
- Documenting High Level Requirements
- Owning and Accepting the Demand, Project, and Initiative

Participates In:

- Intake *
- Deployment *
- Deployment Retro

Business Partner / Requirements Refiner

Responsible For:

- Managing the Portfolio Roadmap
- Engaging Impacted Business Units
- Refining Business Requirements
 - Business Value
 - Value Statement
 - Assumptions
 - Acceptance Criteria
 - Business Dependencies
- Owning and Accepting Requirements

Participates In:

- Intake
- Feature Review
- Deployment Planning *
- Deployment Scrum *
- Defect Triage *
- Deployment *
- Deployment Retro

Business Relationship Manager / Requirements Prioritizer

Responsible For:

- Collaborating across Business Portfolios to stack-rank Demands / Projects based on priority
- Facilitating Intake and documenting LoEs from Impacted Teams
- Facilitating Feature Review and confirming Scope Agreement between Business and IT
- Prioritizing Projects for Deployment Planning

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Deployment *
- Deployment Retro

Release Manager / Deployment Orchestrator

Responsible For:

- Facilitating Deployment Planning and confirming commitment from Impacted Teams
- Facilitating Deployment Scrum to track progress, resolve blocking issues, and validate Deployment Artifacts are accurate
- Facilitating Defect Triage to resolve or mitigate Regression Defects and coordinating Defect Scrums when appropriate
- Attend CAB to represent SCRs that are deploying
- Facilitating the Deployment to validate code in Production, resolve Production Incidents, and communicate completion
- Facilitating the Deployment Retro to identify areas of success, areas of improvement, and take-away action items

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- CAB
- Deployment
- Deployment Retro

Program Manager / Feature Orchestrator

Responsible For:

- Soliciting Go-Live commitments from the Impacted Teams prior to Deployment Planning
- Facilitating collaboration and delivery of Features across Impacted Teams to successfully deploy their Project
- Coordinating Defect Scrums for their Projects

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- CAB *
- Deployment
- Deployment Retro

Deployment Manager / Deployment Driver

Responsible For:

- Ensuring Project commitment during Deployment Planning is based on priority
- Evaluating introduction of new work during the Development Cycle to balance Risk vs. Value and reduce disruptions when possible
- Focusing Impacted Teams on Sev 1 and Sev 2 Defect resolution during Testing Cycle
- Ensuring all Impacted Teams have a representative who is present, informed, and capable of making decisions throughout the Deployment Cycle
- Driving for timely code validation and incident resolution during the Deployment

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- Deployment
- Deployment Retro

Development Manager / Team Leader

Responsible For:

- Ensuring their team is focused on developing the highest priority Features
- Removing any impediments or disruptions to their team during the Development Cycle
- Ensuring their team has a representative who is present, informed, and capable of making decisions throughout the Deployment Cycle

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- Deployment
- Deployment Retro

TQA / Technical Testing

Responsible For:

- Analyzing scope and LoE for proposed projects
- Confirming capacity to test Features at Deployment Planning
- Identifying, Creating, and Executing Manual and Automated Regression and Technical Testing at the Feature level
- Publishing daily TQA status reports during the Test Cycle
- Escalating blocking issues that impede testing
- Verifying successful Test completion satisfies Feature Acceptance Criteria
- Approval / Rejection / Migration of SCRs as appropriate throughout the testing cycle

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- Test Deployment
- Prod Deployment Retro

UAT & DIT / Functional Testing

Responsible For:

- Confirming capacity to test Features at Deployment Planning
- Identifying, Creating, and Executing Manual and Automated Regression and Functional Testing at the Feature level
- Verifying successful Test completion satisfies Feature Acceptance Criteria
- Validating code is performing as expected once deployed to Production

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- Deployment
- Deployment Retro

Product Owner / Feature Driver

Responsible For:

- Collaborating with the Business Partner to convert Business Requirements to Product Features
- Finalizing Feature Scope
- Prioritizing Features in the Product / Platform Backlog
- Refining Features to a 'Dev-Ready' State
- Driving team to develop high quality Features that satisfy the Acceptance Criteria
- Documenting Feature Milestones
- Facilitating Defect resolution

Participates In:

- Intake
- Feature Review
- Deployment Planning
- Deployment Scrum
- Defect Triage
- CAB
- Deployment
- Deployment Retro

Ceremonies

- **Intake / Triage**
- **Feature Review / Joint Planning Session**
- **Deployment Planning Meeting**
- **Deployment Scrum**
- **Defect Triage**
- **CAB**
- **Deployment**
- **Deployment Retro**

Rapid Deployment Ceremony Cadence

Roles	Intake	Feature Review	Deployment Planning	Deployment Scrum	Defect Triage	CAB	Deployment	Deployment Retro
Business Sponsor	Weekly *						Monthly *	Monthly
Business Partner	Weekly	Weekly	Weekly *	Scrum of Scrums 3 X Weekly *	Daily between DTT and Deployment		Monthly *	Monthly
Business Relationship Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly			Monthly *	Monthly
Release Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment	Weekly	Monthly	Monthly
Program Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment	Weekly *	Monthly	Monthly
Deployment Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment		Monthly	Monthly
Development Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment		Monthly	Monthly
TQA Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment		Monthly	Monthly
UAT Manager	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment		Monthly	Monthly
SCM Manager						Weekly	Monthly	Monthly
Product Owner	Weekly	Weekly	Weekly	Scrum of Scrums 3 X Weekly	Daily between DTT and Deployment	Weekly	Monthly	Monthly

* Recommended if appropriate

Intake / Triage

Participants:
(Owner)

- Business Sponsor *
- Business Partner
- **Business Relationship Manager**
- Release Manager
- Program Manager
- Deployment Manager
- Development Manager
- TQA
- UAT & DIT
- Product Owner

* Recommended if appropriate

Entry Criteria:

- Business Case has been created
- Demand has been created in ServiceNow
- Demand has VP Sponsor approval
- Initiative has been created in Rally
- Impacted Teams have at least 8 business hours (between 9:00 AM MST - 4:00 PM MST) to review

Key Activities:

- Business Case reviewed to identify Impacted Teams
- Impacted Teams provide estimated t-shirt size LoE

Exit Criteria / Output:

- Estimated LoE added to Resource Plans on Demand in ServiceNow for Impacted Teams
- Impacted Teams identified on the Initiative in Rally

Metrics

- % of Top 10 Business Ranking Projects
- % EPL (ITPL) Projects
- % of Projects with Resource Plans
- % of Initiatives with Impacted Teams identified

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Feature Review / JPS

Participants:
(Owner)

- Business Partner
- **Business Relationship Manager**
- Release Manager
- Program Manager
- Deployment Manager
- Development Manager
- TQA
- UAT & DIT
- Product Owner

Entry Criteria:

- Requirements have been converted to Features
- Features are documented in the appropriate Business Backlog
- Impacted Teams have reviewed the Features

Key Activities:

- Review each Feature to get Scope Agreement between Business and IT on:
 - Value Statement
 - Assumptions / Pre-conditions
 - Acceptance Criteria
 - Steps to Test
- Identify Research Spikes or Technical Dependencies
- Impacted Teams provide updated LoE

Exit Criteria / Output:

- PO moves Features to Product Backlog in Rally and stack-ranks based on priority
- PO adds Scope Agreement Date to Feature in Rally
- All Features have Acceptance Criteria
- Features are 'Dev-Ready'
- Feature State is 'Design'
- Resources updated in ServiceNow to reflect revised LoE
- Time critical Features have Milestones

Metrics:

- Feature Design Time
- % of Features with Scope Agreement Date at least 6 weeks prior to Deployment

* Recommended if appropriate

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Deployment Planning

Participants: <u>(Owner)</u> <ul style="list-style-type: none">• Business Partner *• Business Relationship Manager• <u>Release Manager</u>• Program Manager• Deployment Manager• Development Manager• TQA• UAT & DIT• Product Owner	Entry Criteria: <ul style="list-style-type: none">• Projects have been reviewed at Intake• Features have been reviewed at Feature Review and have been moved to the Product Backlog and have a Scope Agreement Date• Deployment Backlog stack-ranked based on priority	Key Activities: <ul style="list-style-type: none">• Review <u>Deployment Planning Agenda</u>• PgMs and Impacted Teams commit to Go-Live Milestones based on Value and Capacity• Add notes as appropriate *	Exit Criteria / Output: <ul style="list-style-type: none">• Go-Live Milestones assigned to Projects in ServiceNow• Go-Live Milestones assigned to Features in Rally• Go-Live Milestones added to Deployment Backlog• Cross-reference Projects with Go-Live in ServiceNow to Deployment Backlog and Features in Rally• PgMs communicate changes to previous assignments to all Impacted Teams	Metrics: <ul style="list-style-type: none">• % of Go-Live Milestones assigned to Projects in ServiceNow 6 weeks prior to Deployment• % of Features with Milestones in Rally 6 weeks prior to Deployment
---	---	---	---	---

* Recommended if appropriate

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Deployment Scrum

Participants: <u>(Owner)</u> <ul style="list-style-type: none">● Business Partner● Business Relationship Manager● <u>Release Manager</u>● Program Manager● Deployment Manager● Development Manager● TQA● UAT & DIT● Product Owner	Entry Criteria: <ul style="list-style-type: none">● <u>Deployment Scrum Email</u> sent prior to meeting with <u>RD Target Dates</u> listed● Features, User Stories, Defects, and SCRs are accurate and up to date	Key Activities: <ul style="list-style-type: none">● Review <u>Deployment Scrum Agenda</u>● PgMs provide updates on their Initiatives to include● Review Deployment Artifacts<ul style="list-style-type: none">○ Features○ User Stories○ Defects○ SCRs	Exit Criteria / Output: <ul style="list-style-type: none">● Blocking issues and dependencies documented with individual responsible to resolve identified● Required Scrums or follow up actions communicated with individual responsible identified● All Deployment Artifacts are up-to-date and accurate	Metrics <ul style="list-style-type: none">● Total # of Features with Deployment Milestone● Total # of Stories associated to Features with Deployment Milestone● % of User Stories Accepted by DTT● % of Features with Actual End Date on or before the Deployment
<small>* Recommended if appropriate</small>				

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Drop to Test

Participants: <u>(Owner)</u>	Entry Criteria:	Key Activities:	Exit Criteria / Output:	Metrics
<ul style="list-style-type: none">• Business Partner• Business Relationship Manager• <u>Release Manager</u>• Program Manager• Deployment Manager• Development Manager• TQA• UAT & DIT• Product Owner	<ul style="list-style-type: none">• For BAU Deployments, this will be the Thursday of the week prior to the Deployment• For the monthly Deployments, this will be the last Thursday of the month prior to the Deployment	<ul style="list-style-type: none">• TBD	<ul style="list-style-type: none">• TBD	<ul style="list-style-type: none">• TBD
* Recommended if appropriate				

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Defect Triage *

Participants: (Owner) <ul style="list-style-type: none">• Business Partner• Business Relationship Manager• <u>Release Manager</u>• Program Manager• Deployment Manager• Development Manager• TQA• UAT & DIT• Product Owner	Entry Criteria: <ul style="list-style-type: none">• <u>Defects</u> created in ALM• Defect Triage invite sent to appropriate distro prior to Drop to Test• Impacted Teams have at least 2 business hours (between 9:00 AM MST - 4:00 PM MST) to review defects assigned to their team	Key Activities: <ul style="list-style-type: none">• Review <u>Defect Triage Agenda</u>• 'Assigned To' individual or appropriate representative speaks on behalf of the Defect• Action items and notes are documented as appropriate	Exit Criteria / Output: <ul style="list-style-type: none">• Defect is assigned to the appropriate party who can resolve• Defect status is updated• Scrums are scheduled as needed	Metrics: <ul style="list-style-type: none">• % of Defects Reopened• Average Time to close a Defect• Defect Churn
--	---	--	--	---

* Recommended if appropriate

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Change Advisory Board (CAB)

Participants: (Owner) <ul style="list-style-type: none">• <u>SCM</u>• Release Management• Program Manager *• Business Operations Manager• TQA• UAT & DIT• Product Owner	Entry Criteria: <ul style="list-style-type: none">• SCR has been created following <u>SCR Standards</u> and tagged with the appropriate deployment• To tag an SCR, update the Rapid Deployment Date field with the appropriate Rapid Deployment in the following formant : RD-YYYY.MM.DD• Review email from SCM containing SCRs scheduled for deployment that week• SCRs scheduled for the deployment are in CAB state by 3:00 PM MST the day prior	Key Activities: <ul style="list-style-type: none">• SCM reviews each SCR tagged for that week's deployment• If requested by SCM, tag SCRs with the appropriate Rapid Deployment tag	Exit Criteria / Output: <ul style="list-style-type: none">• All SCRs scheduled for deployment to Production have been reviewed and approved• All SCRs that are part of the RD, have the appropriate RD-tag	Metrics: <ul style="list-style-type: none">• % of SCRs tagged after CAB• % of SCRs not in CAB state prior to CAB• Total number of SCRs deployed to Production
--	---	---	--	--

* Recommended if appropriate

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Deployment

Participants:
(Owner)

- Business Partner
- Business Relationship Manager
- **Release Manager**
- Program Manager
- Deployment Manager
- Development Manager
- TQA
- UAT & DIT
- Product Owner

Entry Criteria:

- All Features in ‘Test’ state and marked ‘Ready’
- All SCRs in appropriate state
- All User Stories are Accepted
- All Defects are all closed
- Appropriate environment clusters are moved out of Production
- Deployment Bridge is set up and sent to appropriate distro

Key Activities:

- Display Release Day Incident Report and Validation Sheet on the Deployment Bridge
- Code is deployed to appropriate cluster
- Code is validated by the appropriate teams that it has correctly been deployed
- Code is tested by UAT / DIT to ensure it is functioning as expected
- Clusters are turned back on in Production

Exit Criteria / Output:

- Successful validation of all code pushed into Production
- Code repositories, Rally, and ServiceNow are updated
- All Incidents are Targeted for Resolution or Resolved

Metrics:

- # of High Severity Production Incidents
- # of Low Severity Production Incidents
- Duration of Deployment Day Bridge
- How many Production Incidents could of prevented in Test

* Recommended if appropriate

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Deployment Retro

Participants: <u>(Owner)</u> <ul style="list-style-type: none">• Business Sponsor *• Business Partner• Business Relationship Manager• <u>Release Manager</u>• Program Manager• Deployment Manager• Development Manager• TQA• UAT & DIT• Product Owner• Dev Team *	Entry Criteria: <ul style="list-style-type: none">• Jamboard set up and included on the invite for the Deployment Retro• Complete and share the <u>Deployment Scorecard</u>• May include the following topics:<ul style="list-style-type: none">○ Areas of Success○ Areas of Opportunity○ Lessons Learned○ Action Items	Key Activities: <ul style="list-style-type: none">• Review Action Items from prior Retro• Each participant contributes at least one item in each category• Each item is presented and reviewed by the individual who submitted the item• Team members vote to indicate they agree with specific items	Exit Criteria / Output: <ul style="list-style-type: none">• Action Items are created for areas of improvement and ownership is assigned to individual responsible for resolving	Metrics <ul style="list-style-type: none">• % of Action Items completed from previous Retro
---	---	---	--	--

* Recommended if appropriate

Intake	Feature Review	Deployment Planning	Deployment Scrum	Drop to Test	Defect Triage	CAB	Deployment	Deployment Retro
--------	----------------	---------------------	------------------	--------------	---------------	-----	------------	------------------

Appendix

- **Feature Template**
- **User Story Template**
- **Glossary**

SDLC Timeline

- **At any point in time on a rolling cycle**
 - 8 - 10 Weeks Prior to Deployment
 - BRMs are working with BP to define and refine Demands and Projects
 - 6 - 8 Weeks Prior to Deployment
 - BP and PO are working to define and refine Features
 - 4 - 6 Weeks Prior to Deployment
 - PO and SM are working to define and refine User Stories
 - 2 - 4 Weeks Prior to Deployment
 - Development Teams are working to complete User Stories
 - 0 - 2 Weeks Prior to Deployment
 - Development Teams and Testing Teams are working to resolve all open Defects

F26586

1

Title - Sample Feature

Details

User Stories

Discussions

Dependencies

Risks

Revision History

STATUS

2

Ready

Blocked

DESCRIPTION

3

Font

Size

B

I

U

S

A

□

↶

↷

↻

↺

↻

↻

↻

↻

Value Statement

- As a USER
- I want to PERFORM AN ACTION
- In order to ACHIEVE DESIRED RESULT

(Example)

- As an EXISTING CUSTOMER
- I want to LOG INTO MY DISH ACCOUNT
- In order to UPDATE MY ADDRESS

Pre-Conditions / Assumptions

- Conditions that must be present before USER may PERFORM AN ACTION

(Example)

- Existing Customer
- Account in Good Standing

Acceptance Criteria

- Outcome or Value the USER will receive, NOT a Solution

(Example)

- EXISTING CUSTOMER enters their new address
- Address is validated
- New address is saved

Steps to Test

- Logical steps USER would take to ACHIEVE DESIRED RESULT

(Example)

- Log in to MyDish
- Access Account Info tab
- Select 'Edit'
- Enter New Address
- Select 'Validate'
- Scrubbed Address is Presented
- Select 'Save'

ATTACHMENTS

17

+ Drag or click to add attachments

NOTES

Font

Size

B

I

U

S

A

□

↶

↷

↻

↺

↻

↻

↻

↻

4

OWNER

Adam Nichols

5

PROJECT

DISH Enterprise

PERCENT DONE BY STORY COUNT

0%

6

STATE

-- No Entry --

7

RELEASE

Unscheduled

8

PLANNED START DATE

select date

9

PLANNED END DATE

select date

10

MILESTONES

+

11

PARENT

+

12

03. PROGRAM MANAGER (PGM)

+

13

05. DEMAND NUMBER

0

14

06. PROJECT NUMBER

15

07. IMPACTED TEAMS

+

16

08. SCOPE AGREEMENT DATE

select date

28

Story Template

US141551

1

Title - Sample User Story

Show Fields

Templates

Details

Tasks

Children

Test Cases

Test Run

Defects

Discussions

Revision History

Charts

Risks

Dependencies

Connections

STATUS

Ready

Blocked

2

DESCRIPTION

3

Font

Size

B

I

U

S

A

A

I

X

Value Statement

As a USER

I want to PERFORM AN ACTION

In order to ACHIEVE DESIRED RESULT

(Example)

As an EXISTING CUSTOMER

I want to PERFORM AN ACTION

In order to ACHIEVE DESIRED RESULT

Pre-Conditions / Assumptions

Conditions that must be present before USER may PERFORM AN ACTION

Test Statement

Scenario - Label for Test Scenario

Given - Beginning state of the Scenario

When - Specific action that the user takes

Then - Testable outcome, typically a result of the action in 'When'

(Example)

GIVEN I have logged into dish.com

WHEN I open my profile page to edit my address

THEN I should be able to update and save my new address

Request / Response

Description of the Request being sent and the expected Response

Steps to Test

Happy Path' series of steps the USER would logically take to ACHIEVE DESIRED RESULT

When passed will result in satisfying the Acceptance Criteria

4

OWNER

Adam Nichols

5

PROJECT *

d.x.ace

6

SCHEDULE STATE *

P

7

PLAN EST

5

8

ITERATION

2020.04.01-2020.04.07

CREATION DATE

2020-02-03

9

FEATURE

F26586: Title - Sample Feature

User Story Writing Guidelines		
Field	Definition	Responsible (Primary / Secondary)
1) Title	a) Brief description of the expected deliverable of the User Story	SM / PO / Dev Team
2) Status *	a) Ready i) Indicates the User Story has met the Exit Criteria of the current State and is Ready to be pulled forward a) Blocked i) Indicates Issues are present which are preventing the User Story from being worked	SM / PO / Dev Team
3) Description	a) Value Statement i) Identifies the target audience and specific action they must complete to receive a specific outcome	SM / PO / Dev Team
	b) Pre-Conditions / Assumptions * i) Specific criteria that must be met in order for the END USER to be eligible to ACHIEVE the DESIRED RESULT	SM / PO / Dev Team
	c) Test Statement * i) Behavior-specific syntax used to define automated test scripts	SM / PO / Dev Team
	c) Request / Response * i) Description of the Request being sent and the expected Response	SM / PO / Dev Team
	d) Steps to Test * i) 'Happy Path' series of steps the USER would logically take to ACHIEVE DESIRED RESULT ii) When passed will result in satisfying the Acceptance Criteria	SM / PO / Dev Team
4) Owner	a) Initially the Creator of the User Story b) Updated to identify individual who will complete the work of the User Story	SM / PO / Dev Team
5) Project	a) Represents the Development Team that will complete the work and the Execution branch (i.e.- d.x.team example)	SM / PO / Dev Team
6) Scheduled State	a) Defined i) Story has been Defined and is in the Backlog ii) Ready' status may be used to indicate the User Story has been prioritized in the Backlog and is ready for development	SM / PO / Dev Team
	b) In Progress i) Development has started	SM / PO / Dev Team
	c) Complete i) Development is complete	SM / PO / Dev Team
	e) Accepted i) Testing in Dev Environment is complete ii) User Story is ready to move to Test	SM / PO / Dev Team
7) Planned Estimate	a) Represent Story Points b) Used to estimate i) Level of Effort ii) Complexity iii) Risk c) Used to measure i) Velocity ii) Capacity	SM / PO / Dev Team
8) Iteration	a) Consistent timeframe during which Development is completed b) May also be called a 'Sprint'	SM / PO / Dev Team
9) Feature	a) Parent Feature that the User Story is a child of	SM / PO / Dev Team

Glossary

- **Acceptance Criteria** - Specific conditions, with clearly defined pass / fail results, that must be met, before a Feature can be Accepted
- **Artifact** - Theme, Initiative, Feature, User Story, or Task
- **Assumption** (*a.k.a. Pre-Condition*) - Specific criteria that must be met before the end user is eligible to achieve the desired outcome identified in the Requirements or Acceptance Criteria
- **Backlog** - Stack-ranked, prioritized list of work items to be completed
 - **Portfolio Backlog** - Project / Initiative artifacts prioritized by the Business Sponsor or Business Partner
 - **Product Backlog** - Feature artifacts prioritized by the Product Owner
 - **Team Backlog** - User Story artifacts prioritized by the Scrum Master
 - **Iteration Backlog** - User Story artifacts for a specific team, for a specific Iteration, prioritized by the Scrum Master
- **Bug** - Error found in the Development environment
- **Capacity** - Maximum amount of work a team is capable of delivering in a given time period
- **Customer Journey Mapping** - Exercise of visually documenting the process a customer goes through to achieve the desired outcome specified in the Requirements or Acceptance Criteria
- **Cycle Time** - Time between when work begins on an Artifact and when work is complete on an Artifact
- **Defect** - Error found in the Test environment
- **Definition of Done** - All Acceptance Criteria has been met, Testing is complete, code is deployed into Production, and all Artifacts are updated
- **Demand** - TBD
- **Dependencies** - Completion of one Artifact is reliant upon the completion of another Artifact
- **Deployment** - Artifact has met the Definition of Done and is deployed into the Production environment

Glossary

- **Dev-Ready** - Both the Business and IT agree upon the Value Statement, Scope, and Acceptance Criteria of an Artifact and there is sufficient detail for development to begin
- **Feature** - Independently deployable functionality that satisfies the Definition of Done for the Requirement
- **Impacted Teams** - Individual teams that will be required to complete work in order to meet the Definition of Done for the Artifact
- **Incident** - Error found in Production
- **Initiative** - Group of one or many Features that collectively deliver a specific outcome for the end user
- **Iteration** (*a.k.a. - Sprint*) - Pre-defined consistent time box, during which Impacted Teams develop and deliver User Stories, typically between 1 - 3 weeks
- **Lead Time** - Time between when an Artifact is created and when work is complete on an Artifact
- **Level of Effort** (*LoE*) - Estimated time (+/- 25%) required to deliver an Artifact
- **Milestone** - Used to track the completion of a set of functionality
- **Project** - TBD
- **Release** - Pre-defined consistent time box, during which Impacted Teams develop and deliver Features, typically between 1 month - 1 quarter
- **Requirement** - Desired outcome, output, or functionality requested by the Business on behalf of the End User
- **Return on Investment** (*RoI*) - Benefit (*Sales / Savings*) divided by the Expense (*Labor / Resources*) required to deliver an Artifact
- **Roadmap** - Strategic plan that identifies the key priorities major milestones required to deliver specified outcomes
 - **Portfolio Roadmap** - 6 - 18 month plan that identifies critical Business outcomes, typically at the Project or Initiative level
 - **Product Roadmap** - 1 - 6 month plan that identifies critical Product outcomes, typically at the Feature level
- **Roadmap** - Desired outcome or functionality that a specific Artifact will deliver

Glossary

- **Software Change Request (SCR)** - TBD
 - **Spike** - A type of User Story used to track the work to research a solution to a Feature or Story when the Impacted Team does not know how to meet the Definition of Done
 - **Story Point** - Abstract metric used to measure the complexity, risk, and time required to deliver a User Story. Is used as the baseline to determine a team's Capacity and Velocity. Modified Fibonacci Sequence is one of the more popular methods
 - **Task** - Smallest unit of work, owned by an individual, and typically no smaller than 2 hours and no larger than 4 hours
- **Test** - Ensure the execution of the Artifact satisfies the Acceptance Criteria and Steps to Test
 - **Automated** - Test cases are executed automatically with the assistance of a software tool
 - **Manual** - Tests cases executed manually by an individual
 - **Regression** - Ensure that deployment of new Feature does not negatively impact existing functionality
 - **Functional** - Ensure the deployment of new Feature satisfies the functional specifications of the Feature
 - **User Story** - Unit of work for a specific Impacted Team that satisfies specific Acceptance Criteria within a Feature, completed within one Iteration

Glossary

- **Value** - Objective measure used to prioritize an Artifact based on:
 - Business Rank
 - Return on Investment
 - Dependencies
 - Legal Requirements
 - Market Conditions
 - Other Factors
- **Value Statement** - Describes the intended outcome or value received by the end user as a result of a specific action
 - As a _____(*End User*)
 - I want to _____(*Action Performed*)
 - In order to _____(*Outcome / Value Received*)
- **Velocity** - Rolling average number of Artifacts delivered over a given number of Iterations (*User Stories*) or Releases (*Features*)