

A hand-drawn diagram of a soccer field in the background, featuring white 'X' marks for players, white circles for the ball, and white arrows indicating movement paths.

ACE Playbook 2.0

Index

Overview:

- Software Delivery Framework
- Roles
- Responsibility Matrix
- Ceremonies
 - Ceremony Cadence
- Appendix
 - Glossary
 - Templates
 - Story Pointing

Software Delivery Framework

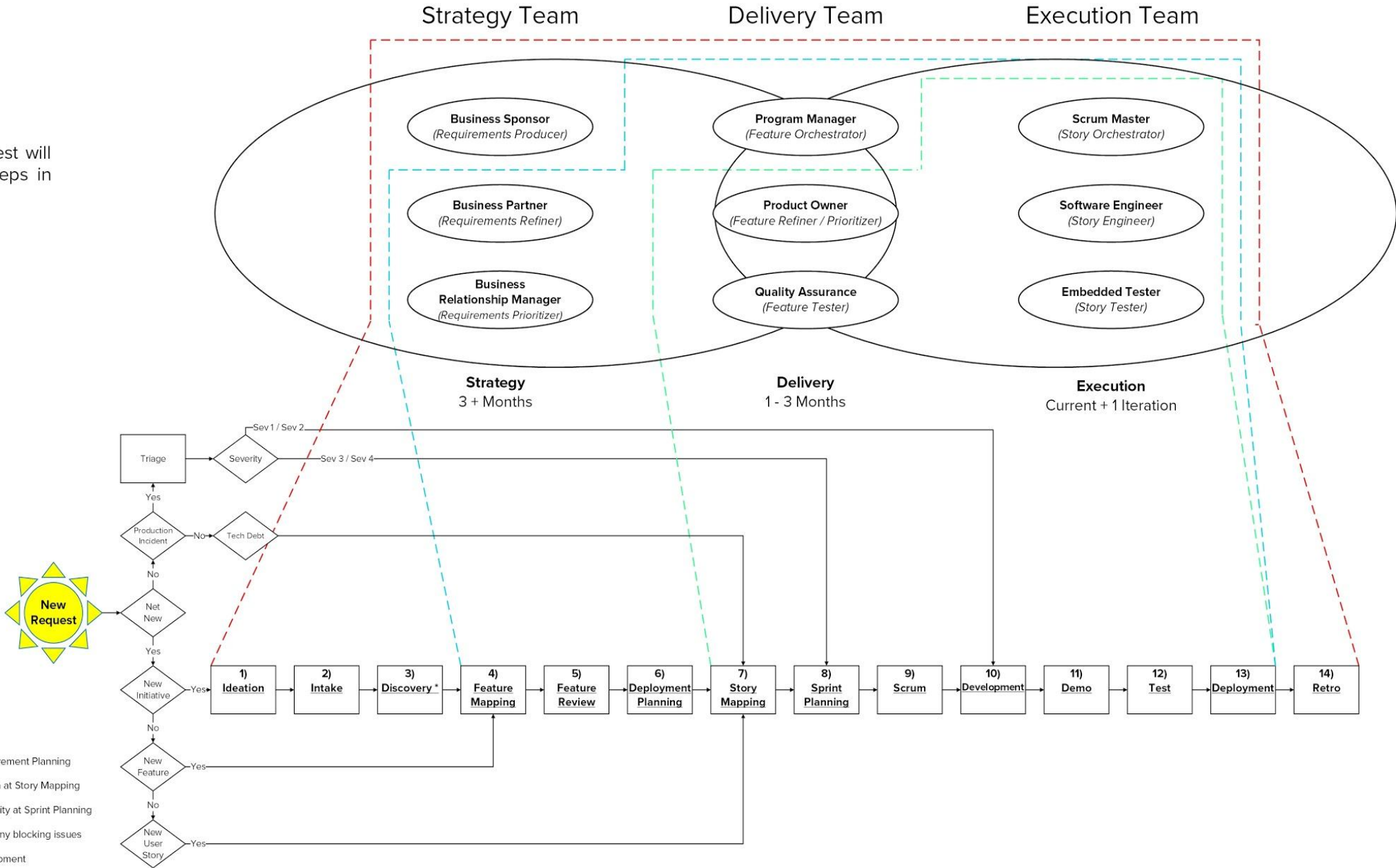
The type, size, and complexity of a request will determine who is impacted and what steps in the process need to be followed

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
- **Product Manager (PdM)** / **Business Partner (BP)** / Requirements Refiner
- **Business Relationship Manager (BRM)** / Requirements Prioritizer
- **Program Manager (PgM)** / Feature Orchestrator
- **Product Owner (PO)** / Feature Refiner & Prioritizer
- **Quality Assurance (QA)** / Feature Tester
- **Scrum Master (SM)** / Story Orchestrator
- **Software Engineer (SE)** / Story Engineer
- **Embedded Tester (ET)** / Story Tester

Responsibility Matrix

ROLES R = Responsible P = Participates	Ideation			Refinement			Development						Test			Deploy	
	Define Initiative	Refine Initiative	Prioritize Initiative	Define Feature	Refine Feature	Prioritize Feature	Define User Story	Refine User Story	Prioritize User Story	Develop User Story	Test User Story	Accept User Story	Test Feature	Demo Feature	Accept Feature	Deploy	Retro
Requirements Producer	R	P*	P*	P*										P	P*		
Requirements Refiner	P*	R	P*	P	P*	P								P*	R		
Requirements Prioritizer	P	P	R	P*	P*	P*								P*			
Feature Orchestrator			P*	P*	P*	P*			P*					P*		P	
Feature Refiner / Prioritizer		P*	P*	R	R	R	P*	P	P*					R	P	P	
Feature Tester				P*	P	P	P*	P*	P*				R	P*		P*	
Story Orchestrator					P	P	R	R	R	P*	P*	R	P*			R	R
Story Engineer							P*	P	P	R	P	P*	P*			P*	P
Story Tester							P	P	P	P	R	P*	P*	P*			P

* Recommended if appropriate

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:

- Ideation
- Intake *
- Discovery *
- Demo *

Business Partner / Product Manager / 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:

- Ideation
- Intake
- Discovery
- Feature Mapping
- Feature Review
- Deployment Planning *
- Scrum
- Demo
- Deployment *
- Retro *

Business Relationship Manager / Requirements Prioritizer

Responsible For:

- Collaborating across Lines of Business and Business Portfolios to stack-rank Business Requirements based on:
 - Business Value
 - Level of Effort
 - Return on Investment
 - Dependencies
 - Legal / Contractual Requirements
 - Other Factors

Participates In:

- Ideation *
- Intake
- Discovery *
- Feature Review
- Deployment Planning
- Scrum
- Demo *
- Retro *

Program Manager / Feature Orchestrator

Responsible For:

- Facilitating collaboration and delivery of Features across:
 - Lines of Business
 - Portfolios
 - Products
 - Platforms
 - Applications
 - Development Teams

Participates In:

- Intake
- Discovery *
- Feature Mapping *
- Feature Review
- Deployment Planning
- Scrum
- Demo
- Test
- Deployment
- Retro *

Product Owner / Feature Refiner & Prioritizer

Responsible For:

- Establishing and Managing the Product / Platform Roadmap
- 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
- Identifying Feature Milestones

Participates In:

- Intake
- Discovery
- Feature Mapping
- Feature Review
- Deployment Planning
- Story Mapping *
- Sprint Planning *
- Scrum
- Development *
- Demo
- Test *
- Deployment
- Retro

Quality Assurance / Feature Tester

Responsible For:

- Identifying, Creating, and Executing Manual and Automated Regression and Functional Testing at the Feature level
- Verifying successful Test completion satisfies Feature Acceptance Criteria

Participates In:

- Intake
- Discovery *
- Feature Mapping *
- Feature Review
- Deployment Planning
- Scrum
- Demo
- Test
- Deployment
- Retro

Scrum Master / Story Orchestrator

Responsible For:

- Breaking down Features into User Stories
- Establishing and Managing the Sprint Backlog
- Removing Obstacles, Impediments, and Distractions from the Sprint
- Facilitating cross-team collaboration
- Communicating progress and completion status to Feature Level
- Driving Agile Best Practices

Participates In:

- Discovery *
- Feature Mapping
- Feature Review
- Deployment Planning
- Story Mapping
- Sprint Planning
- Scrum
- Development
- Demo
- Test
- Deployment
- Retro

Software Engineer / Story Engineer

Responsible For:

- Developing functional software that satisfies the Acceptance Criteria defined in the User Story
- Identifying, Documenting, and Completing Technical Debt and Story Spikes
- Developing and Executing Unit Tests

Participates In:

- Feature Review *
- Story Mapping
- Sprint Planning
- Scrum
- Development
- Demo
- Test
- Deployment *
- Retro

Embedded Tester / Story Tester

Responsible For:

- Identifying and Documenting Bugs and Defects
- Identifying, Creating, and Executing Manual and Automated Testing at the User Story level

Participates In:

- Feature Review *
- Story Mapping
- Sprint Planning
- Scrum
- Development
- Demo
- Test
- Deployment *
- Retro

Ceremonies

- Ideation
- Intake
- Discovery
- Feature Mapping
- Feature Review
- Deployment Planning
- Story Mapping
- Sprint Planning
- Scrum
- Development
- Demo
- Test
- Deployment
- Retro

Ceremony Cadence

Roles	Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro
Business Sponsor	Monthly	Weekly *	As Needed *								End of Sprint *			
Business Partner / Product Manager	Monthly	Weekly	As Needed	Weekly	Weekly	Weekly *			Scrum of Scrums 3 X Weekly *		End of Sprint		Weekly *	Weekly *
Business Relationship Manager	Monthly *	Weekly	As Needed *		Weekly	Weekly			Scrum of Scrums 3 X Weekly *		End of Sprint *			Weekly *
Program Manager		Weekly	As Needed *	Weekly *	Weekly	Weekly			Scrum of Scrums 3 X Weekly		End of Sprint	Daily	Weekly	Weekly *
Product Owner		Weekly	As Needed	Weekly	Weekly	Weekly	Weekly *	Day Before Sprint Begins *	Scrum of Scrums 3 X Weekly	Daily *	End of Sprint	Daily *	Weekly	Weekly
Quality Assurance		Weekly	As Needed *	Weekly *	Weekly	Weekly			Scrum of Scrums 3 X Weekly		End of Sprint	Daily	Weekly	Weekly
Scrum Master			As Needed	Weekly	Weekly	Weekly	Weekly	Day Before Sprint Begins	Scrum of Scrums 3 X Weekly + Daily Scrum	Daily	End of Sprint	Daily	Weekly	Weekly
Software Engineer					Weekly *		Weekly	Day Before Sprint Begins	Daily Scrum	Daily	End of Sprint	Daily	Weekly	Weekly
Embedded Tester					Weekly *		Weekly	Day Before Sprint Begins	Daily Scrum	Daily	End of Sprint	Daily	Weekly	Weekly

Ideation

Participants / (Owner) <ul style="list-style-type: none">• <u>Business Sponsor</u>• Business Partner / Product Manager• Business Relationship Manager *	Entry Criteria: <ul style="list-style-type: none">• Request has been submitted	Key Activities: <ul style="list-style-type: none">• Create Business Case• Define Project Value• Submit Demand in ServiceNow	Exit Criteria / Output: <ul style="list-style-type: none">• Demand created in ServiceNow• Demand submitted for VP Sponsor Approval• Demand communicated to BRM	Metrics: <ul style="list-style-type: none">• Initiative Lead Time
--	---	--	---	--

* Recommended if appropriate

Intake

Participants: (Owner)

- Business Sponsor *
- Business Partner / Product Manager
- Business Relationship Manager
- Program Manager
- Product Owner
- Quality Assurance

Entry Criteria:

- Business Case has been created
- Demand has VP Sponsor approval

Key Activities:

- Business Case reviewed to identify potentially Impacted Teams
- Impacted Teams (including Business Units) 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

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro	18
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------	----

Discovery *

Participants: (Owner)

- Business Sponsor *
- **Business Partner / Product Manager**
- Business Relationship Manager *
- Program Manager *
- Product Owner
- Quality Assurance *
- Scrum Master

Entry Criteria:

- Impacted Teams (including Business Units) have reviewed the Business Case

Key Activities:

- Conduct Customer Journey Mapping session (Appendix)
 - Customer segments and touch points identified
- Review Product Risks
 - Value Risk
 - Usability Risk
 - Feasibility Risk
 - Business Viability Risk
- Business Requirements / Outcomes clarified and refined
- Business Requirements / Outcomes stack-ranked based on priority

Exit Criteria / Output:

- Business Requirements / Outcomes documented as Features at the Business branch in Rally stack-ranked based on priority
- Impacted 'Decision Makers' identified
- Potential high level Features identified
- Submit Demand for Development Approval
- Customer Journey Map or Customer Journey Flow

Metrics:

- TBD

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------

Feature Mapping

Participants: (Owner) <ul style="list-style-type: none">• Business Partner / Product Manager• Program Manager *• <u>Product Owner</u>• Quality Assurance *• Scrum Master	Entry Criteria: <ul style="list-style-type: none">• Demand has been approved for Development and converted to a Project in ServiceNow• Business Requirements stack-ranked based on priority	Key Activities: <ul style="list-style-type: none">• Functionally identify the unique steps required to deliver the outcome of each Business Requirement• Differentiate between new or existing functionality• Identify Impacted Teams required to develop and deliver new functionality• Identify functional dependencies between Features or Impacted Teams• Ensure all Security and Accessibility Compliance Standards and Guidelines are met	Exit Criteria / Output: <ul style="list-style-type: none">• Documented Features at the Business branch in Rally that deliver the desired outcome for each Business Requirement• Impacted Teams identified on the Features in Rally	Metrics <ul style="list-style-type: none">• Feature Lead Time• % of Features created at least 6 weeks prior to Deployment• % of Requirements Defects compared to number of Features Deployed
--	---	--	--	---

* Recommended if appropriate

Feature Review

Participants: (Owner)

- Business Partner / Product Manager
- Business Relationship Manager
- Program Manager *
- Product Owner
- Quality Assurance
- Scrum Master
- Software Engineer *
- Embedded Tester *

Entry Criteria:

- Impacted Teams have reviewed Features at the Business branch

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 provided updated LoE

Exit Criteria / Output:

- Features stack-ranked based on priority in the Product Backlog in Rally
- All Features have Acceptance Criteria
- Scope Agreement Date added to Feature in Rally
- Features are 'Dev-Ready'
- Feature State is 'Backlog'
- 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 4 weeks prior to Deployment

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro	21
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------	----

Deployment Planning

Participants: (Owner) <ul style="list-style-type: none">• Business Partner / Product Manager *• <u>Business Relationship Manager</u>• Program Manager• Product Owner• Quality Assurance• Scrum Master	Entry Criteria: <ul style="list-style-type: none">• Projects have been reviewed at Intake• Projects have gone through Discovery, if required• Features have been reviewed at Feature Review• Time critical Features have Milestones• Projects have Planning Milestones in ServiceNow• Project Backlog stack-ranked based on priority	Key Activities: <ul style="list-style-type: none">• Review each Project and / or Feature based on Business Ranking, Time Criticality, Priority List, LoE, and Dependencies• Impacted Teams commit to Go-Live Milestones based on Value and Capacity	Exit Criteria / Output: <ul style="list-style-type: none">• Deployment Milestones assigned to Features in Rally• Go-Live Milestones assigned to Projects in ServiceNow• Changes to previous assignments are approved by and communicated to all Impacted Stakeholders and 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

Story Mapping

Participants: (Owner)

- Product Owner *
- **Scrum Master**
- Software Engineer
- Embedded Tester

Entry Criteria:

- Refined, Prioritized Product Backlog with Features in a 'Dev-Ready' state

Key Activities:

- Identify (technical) User Stories required to satisfy Feature Acceptance Criteria
- Create User Stories using: *
 - Value Statement
 - Gherkin Method
 - Acceptance Criteria
 - Steps to Test
 - Request / Response

Exit Criteria / Output:

- Product Owner, Quality Assurance, and Scrum Master agree that completion of all child User Stories will satisfy Feature Acceptance Criteria

Metrics:

- TBD

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------

Iteration Planning

Participants: (Owner) <ul style="list-style-type: none">• Product Owner *• <u>Scrum Master</u>• Software Engineer• Embedded Tester	Entry Criteria: <ul style="list-style-type: none">• Prioritized User Story Backlog	Key Activities: <ul style="list-style-type: none">• Review previous Iterations to determine Velocity• Review current and future Iterations to determine capacity• Pull list of User Stories, based on priority, the team may be able to complete in next Iteration• Review each User Story and get consensus from team on Story Points	Exit Criteria / Output: <ul style="list-style-type: none">• Based on capacity, team commits to delivering specific User Stories in the next Iteration• User Stories are stack-ranked in the Iteration Backlog based on priority	Metrics: <ul style="list-style-type: none">• % of User Stories with an Iteration Assigned• % of User Stories with Planned Estimate• % of Scope Creep during an Iteration
---	---	--	---	---

* Recommended if appropriate

Scrum

Participants: (Owner)

Scrum of Scrums

- Business Partner / Product Manager *
- Business Relationship Manager
- **Program Manager**
- Product Owner
- Quality Assurance
- Scrum Master

Daily Scrum

- **Scrum Master**
- Software Engineer
- Embedded Tester

Entry Criteria:

- Participants have access to current accurate data

Key Activities:

- Review what has been completed since previous Scrum
- Review what will be completed before next Scrum
- Review blocking issues and dependencies

Exit Criteria / Output:

- Program Manager / Scrum Master has a list of blocking issues to resolve

Metrics:

- # of Blocked Artifacts

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro	25
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------	----

Development

Participants:
(Owner)

- Product Owner *
- Scrum Master
- **Software Engineer**
- **Embedded Tester**

Entry Criteria:

- User Stories are stack-ranked in the Iteration Backlog based on priority

Key Activities:

- Software Engineers develop code to satisfy the Acceptance Criteria for each User Story
- Unit Tests are passed for each User Story
- Story Testers create manual and / or automated test scripts for each User Story
- Story Engineers report blocking issues to Story Orchestrator, who is responsible for resolving
- Story Orchestrator mitigates interference and distractions for Story Engineers

Exit Criteria / Output:

- User Stories are complete, tested, and Accepted
- All Features are complete, tested, and ready for Production
- All Defects are ‘Closed Complete’

Metrics:

- Feature Cycle Time
- US Cycle Time
- Iteration improvement in Velocity
- % of Unit Test
- % of Automated Testing for US
- # of Code Defects vs. # of US
- % of US started 4 weeks prior to Deployment
- % of US Accepted before Drop to Test
- % of SCRs created and tagged after DTT
- % of SCRs pushed through BAU, Gated, or 1-Click Pipelines

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro	26
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------	----

Demo

Participants: (Owner) <ul style="list-style-type: none">● Business Sponsor *● Business Partner● Business Relationship Manager *● Program Manager● <u>Product Owner</u>● Quality Assurance● Scrum Master● Software Engineer● Embedded Tester	Entry Criteria: <ul style="list-style-type: none">● Feature in ‘Test’ state and all child User Stories in ‘Accepted’ state● Feature and all child User Stories have successfully passed testing	Key Activities: <ul style="list-style-type: none">● Live, working demo of the Feature functionality is presented to the Business ‘Decision Maker’ and Impacted Stakeholders for approval● Based on feedback, decision is made:<ul style="list-style-type: none">○ Ready to Move to Production, no changes required○ Ready to Move to Production, changes to be deployed at a later time through a new Feature○ Not Ready to Move to Production, additional Features and User Stores to be created	Exit Criteria / Output: <ul style="list-style-type: none">● Approved Features are staged for Production	Metrics: <ul style="list-style-type: none">● TBD
---	---	--	--	---

* Recommended if appropriate

Test

Participants: (Owner)

- Program Manager
- Product Owner *
- **Quality Assurance**
- Scrum Master
- Software Engineer
- Embedded Tester

Entry Criteria:

- All Development User Stories have been Accepted
- Features are in 'Test' state in Rally
- SCRs created, tagged to the Deployment, and in 'TQA Approval' or 'TQA / UAT Approval' state in ServiceNow

Key Activities:

- TQA and UAT complete automated and / or manual regression and functional Feature testing
- Defects created in ALM and assigned to appropriate Dev Work Group
- Dev Work Groups analyse, develop, and deploy code to fix the identified Defect

Exit Criteria / Output:

- 100% Run and Pass for all Test Scripts for DIT, TQA, and UAT
- SCRs in 'Deployment Manager Approval' state in ServiceNow
- Features are complete, tested, and ready for Production

Metrics:

- % of Automated Testing for Features
- # of Days to Close a Defect
- % of Test Defects compared to total Defects identified during Test Cycle
- % of Preventable Production Day Incidents

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------

Deployment

Participants: (Owner)

- Business Partner / Product Manager
- **Program Manager**
- Product Owner
- Quality Assurance
- Scrum Master
- Software Engineer
- Embedded Tester *

Entry Criteria:

- Features in 'Test' state and marked 'Ready'
- SCRs in appropriate state

Key Activities:

- Appropriate environment clusters are moved out of Production
- Code is deployed to appropriate cluster
- Clusters are turned back on in Production
- Code is verified in Production

Exit Criteria / Output:

- Successful validation of all code pushed into Production
- Code repositories, Rally, and ServiceNow are updated

Metrics:

- # of High Severity Production Incidents
- # of Low Severity Production Incidents
- Duration of Deployment Day Bridge

* Recommended if appropriate

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro	29
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------	----

Retro

Participants: (Owner)

- Business Partner / Product Manager *
- Business Relationship Manager *
- Program Manager *
- Product Owner
- Quality Assurance
- Scrum Master
- Software Engineer
- Embedded Tester

Participants and Owner determined based on scope of Retro

* Recommended if appropriate

Entry Criteria:

- Entry Criteria and level of detail based on scope of Retro
- Examples:
 - Team Retro = Post Iteration
 - Product Retro = Post Feature Deployment
 - Project Retro = Post Initiative Deployment
 - Deployment Retro = Post Milestone Deployment

Key Activities:

- Review open items from prior Retro
- Team discusses what went well, what was acceptable, and what could be improved
- Each team member is expected to contribute at least one item in each category
- Team members may use checks to indicate they agree with another's input
- Each item is presented and reviewed by the individual who submitted the item

Exit Criteria / Output:

- Action items are created for areas of improvement and assigned to team members to follow up
- Action Item list that is shared with all team members

Metrics

- % of Action Items completed from previous Retro

Ideation	Intake	Discovery	Feature Mapping	Feature Review	Deployment Planning	Story Mapping	Sprint Planning	Scrum	Development	Demo	Test	Deployment	Retro	30
----------	--------	-----------	-----------------	----------------	---------------------	---------------	-----------------	-------	-------------	------	------	------------	-------	----

Appendix

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

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
- **Business Case** - Issue or opportunity to be solved and expected outcome the solution will provide
- **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** - Regardless of the specific Artifact, what must be completed for any Artifact of that type to be considered 'Done'
- **Deliverable** - Desired outcome or functionality that a specific Artifact will deliver
- **Dependencies** - Completion of one Artifact is reliant upon the completion of another Artifact

Glossary

- **Deployment** - Artifact has met the Definition of Done and is deployed into the Production environment
- **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
- **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

Glossary

- **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
 - **Steps to Test** - 'Happy Path' series of steps the *USER* would logically take to *ACHIEVE DESIRED RESULT*, that when passed will result in satisfying the Acceptance Criteria
 - **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
 - **Technical Debt** - Rework incurred for the future due to implementation of the easier, but not the best solution in the present
- **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*)

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

□

↶

↷

↻

↺

↻

↻

↻

↻

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

+

CREATION DATE

2019-12-13

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

36

Story Template

US141551

1

Title - Sample User Story

Show Fields

Templates

Details

Tasks

Children

Test Cases

Test Run

Defects

Discussions

Revision History

Charts

Risks

DependenciesConnections

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