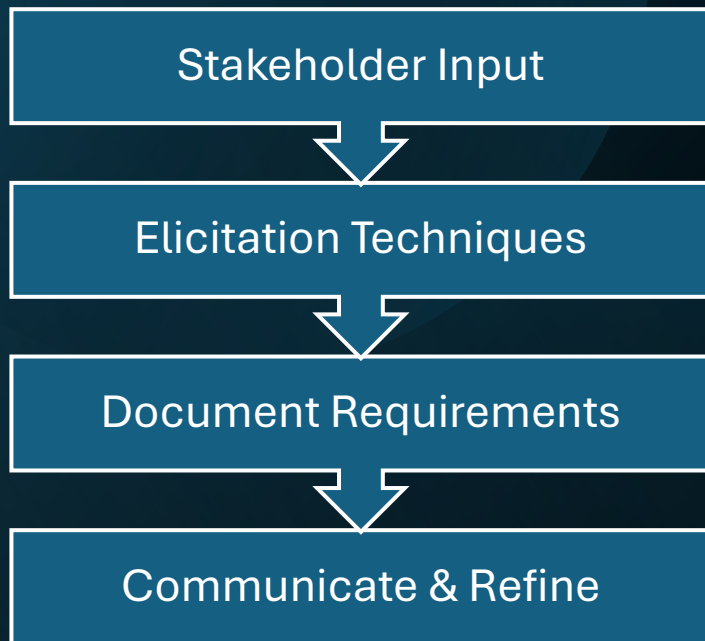# SYS.1 Requirement Elicitation

Introduction to Requirements Elicitation

Requirements elicitation ensures that stakeholder needs are clearly captured and understood by the project team.

1. Gathering Stakeholder Needs

- Requirements shall be gathered through structured techniques like interviews and workshops.
- User stories and job stories capture detailed stakeholder needs.
- Ensure inputs reflect both functional and non-functional expectations.
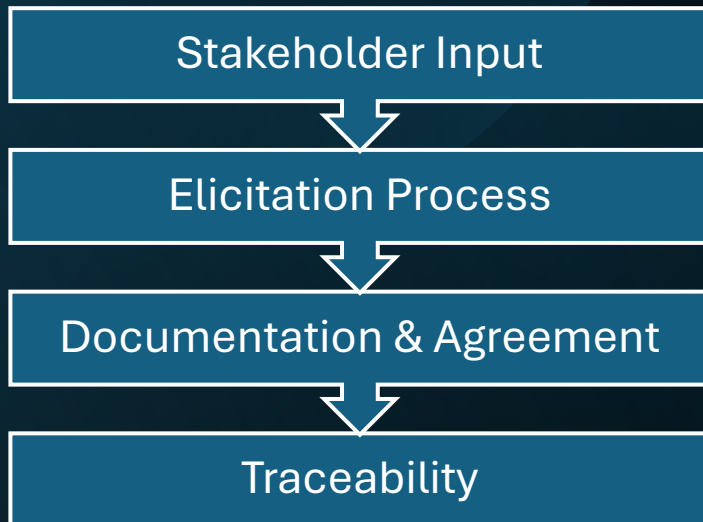
2. Formalizing Requirements

- Stakeholder input shall be converted into formal requirements.
- Use clear formats like user stories and job stories to document.
- Requirements must align with system constraints and goals.

3. Continuous Communication

- Regular communication with stakeholders is critical for refining requirements.
- Keep track of evolving needs throughout the project lifecycle.
- Document changes to ensure they are properly communicated.

Stakeholder Input
↓
Elicitation Techniques
↓
Document Requirements
↓
Communicate & Refine

ASPICE and Requirements Elicitation

Stakeholder Input

↓

Elicitation Process

↓

Documentation & Agreement

↓

Traceability

ASPICE structures the Requirements Elicitation process to ensure alignment and traceability

1. ASPICE Framework Overview

- ASPICE provides structured software development processes.
- SYS.1 focuses on eliciting stakeholder requirements systematically.
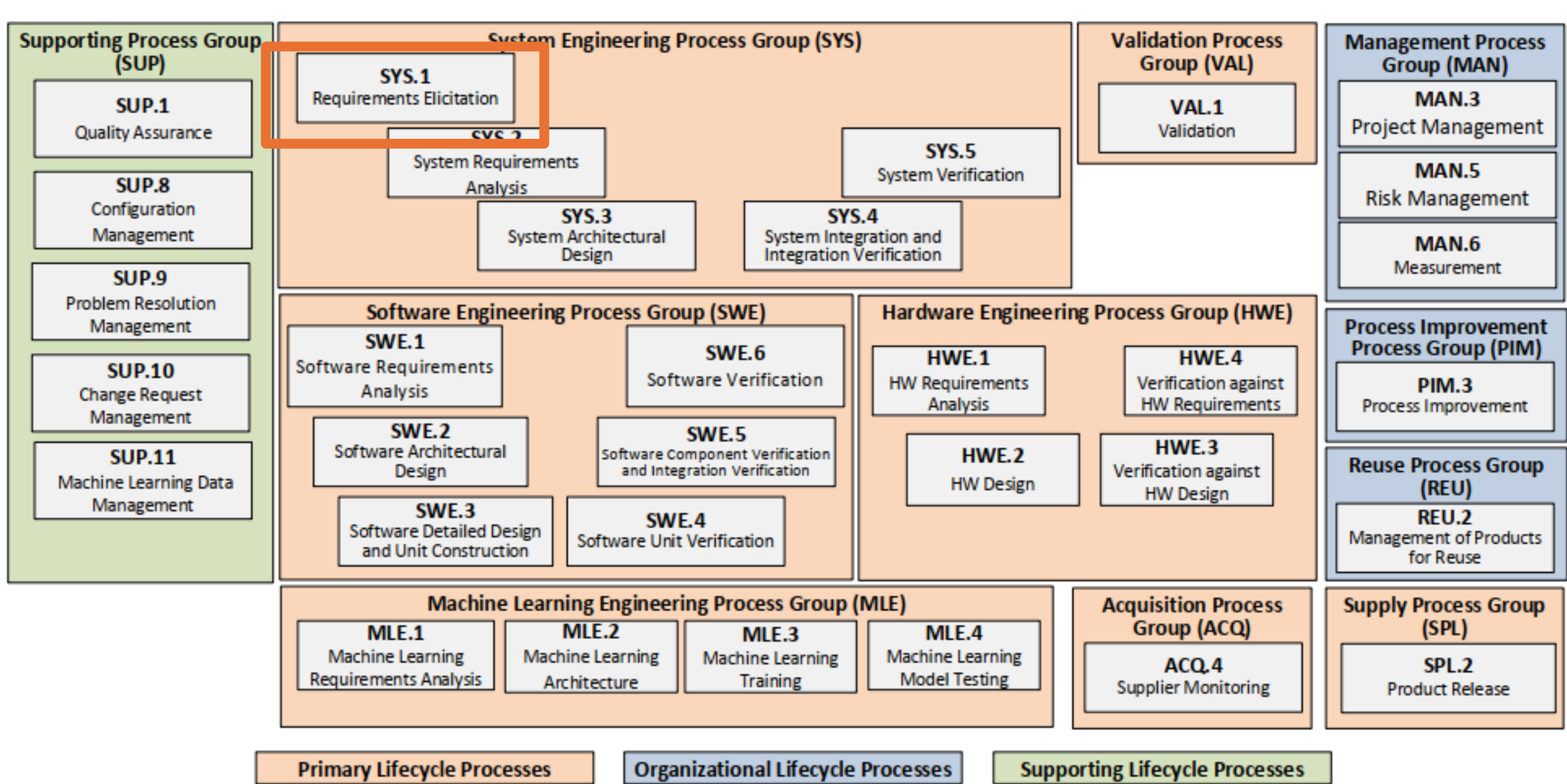- Aligns elicitation with industry standards for traceability.

2. SYS.1 Requirements Elicitation Process

- Steps to gather and document stakeholder needs efficiently.
- Continuous communication and alignment with all stakeholders.
- Tracks requirements from start to project completion.

3. Ensuring Alignment and Traceability

- Promotes traceability from requirements to system design.
- Ensures requirements align with project objectives and goals.
- Reduces risks by maintaining quality and consistency.

Automotive **S**oftware **P**rocess **I**mprovement and **C**apability d**E**termination

Supporting Process Group (SUP)

- **SUP.1** Quality Assurance
- **SUP.8** Configuration Management
- **SUP.9** Problem Resolution Management
- **SUP.10** Change Request Management
- **SUP.11** Machine Learning Data Management

System Engineering Process Group (SYS)

- **SYS.1** Requirements Elicitation
- **SYS.2** System Requirements Analysis
- **SYS.3** System Architectural Design
- **SYS.4** System Integration and Integration Verification
- **SYS.5** System Verification

Validation Process Group (VAL)

- **VAL.1** Validation

Management Process Group (MAN)

- **MAN.3** Project Management
- **MAN.5** Risk Management
- **MAN.6** Measurement

Software Engineering Process Group (SWE)

- **SWE.1** Software Requirements Analysis
- **SWE.2** Software Architectural Design
- **SWE.3** Software Detailed Design and Unit Construction
- **SWE.4** Software Unit Verification
- **SWE.5** Software Component Verification and Integration Verification
- **SWE.6** Software Verification

Hardware Engineering Process Group (HWE)

- **HWE.1** HW Requirements Analysis
- **HWE.2** HW Design
- **HWE.3** Verification against HW Design
- **HWE.4** Verification against HW Requirements

Process Improvement Process Group (PIM)

- **PIM.3** Process Improvement

Reuse Process Group (REU)

- **REU.2** Management of Products for Reuse

Machine Learning Engineering Process Group (MLE)

- **MLE.1** Machine Learning Requirements Analysis
- **MLE.2** Machine Learning Architecture
- **MLE.3** Machine Learning Training
- **MLE.4** Machine Learning Model Testing

Acquisition Process Group (ACQ)

- **ACQ.4** Supplier Monitoring

Supply Process Group (SPL)

- **SPL.2** Product Release

Primary Lifecycle Processes   **Organizational Lifecycle Processes**   **Supporting Lifecycle Processes**

ASPICE – **A**utomotive **S**oftware **P**rocess **I**mprovement and **C**apability d**E**termination

Process ID				
SYS.1				
Process name				
Requirements Elicitation				
Process purpose				
The purpose is to gather, analyze, and track evolving stakeholder needs and requirements throughout the lifecycle of the product and/or service to establish a set of agreed requirements.				
Process outcomes				
1) Continuing communication with the stakeholder is established.				
2) Stakeholder expectations are understood and requirements are defined and agreed.				
3) Stakeholder requirements changes arising from stakeholder needs are analyzed to enable associated risk assessment and impact management.				
4) Determination of stakeholder requirements status is ensured for all affected parties.				

SYS.1 Requirements Elicitation	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
15-51 Analysis Results			X	
13-52 Communication Evidence	X	X		
17-00 Requirement		X		
17-54 Requirement Attribute		X	X	X
Base Practices				
BP1: Obtain stakeholder expectations and requests	X			
BP2: Understand stakeholder expectations		X		
BP3: Agree on requirements		X		
BP4: Analyze stakeholder requirements changes			X	
BP5: Communicate requirements status	X			X

Base practices

SYS.1.BP1: Obtain stakeholder expectations and requests. Obtain and define stakeholder expectations and requests through direct solicitation of stakeholder input, and through review of stakeholder business proposals (where relevant) and other documents containing inputs to stakeholder requirements, and consideration of the target operating and hardware environment.

Note 1: Documenting the stakeholder, or the source of a stakeholder requirement, supports stakeholder requirements agreement and change analysis (see BP3 and BP4).

SYS.1.BP2: Understand stakeholder expectations. Ensure that all affected parties understand each requirement in the same way.

Note 2: Examples of affected parties are customers, suppliers, design partners, joint venture partners, or outsourcing parties.

SYS.1.BP3: Agree on requirements. Formalize the stakeholder's expectations and requests into requirements and obtain an explicit agreement from all affected parties to work on these requirements.

Note 3: The agreed stakeholder requirements may be based on feasibility studies and/or cost and schedule impact analysis.

SYS.1.BP4: Analyze stakeholder requirements changes. Analyze and manage all changes made to the stakeholder requirements against the agreed stakeholder requirements. Assess the impact and risks and initiate appropriate change control and mitigation actions.

Note 4: Requirements changes may arise from different sources as for instance changing technology, stakeholder needs, or legal constraints.

Note 5: Refer to SUP.10 Change Request Management, if required.

SYS.1.BP5: Communicate requirements status. Ensure all affected parties can be aware of the status and disposition of their requirements including changes and can communicate necessary information and data.

SYS.1 Requirements Elicitation Process Overview
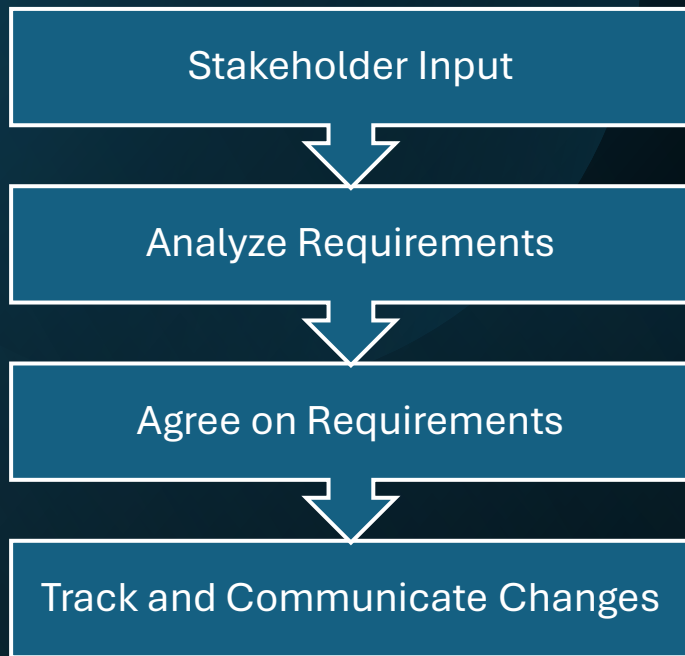
SYS.1 guides the elicitation of requirements from initial gathering to ongoing tracking and communication

1. Purpose of Elicitation

- Collect and document **evolving** stakeholder needs throughout the system's lifecycle.
- Formalize requirements for alignment with system design.
- Keep communication channels open with all stakeholders.

2. Process Outcomes

- Elicited requirements shall be analyzed, agreed upon, and tracked for changes.
- Risk assessment must be performed for evolving requirements.
- Establish agreements on requirements status to prevent conflicts.

3. Impact on Environment

- Continuous elicitation helps capture evolving needs during the system's development.
- Track requirement status from initial agreement to final implementation.
- Ensure that all updates are reflected in project documentation.

Stakeholder Input

Analyze Requirements

Agree on Requirements

Track and Communicate Changes

BP1:
Obtain Stakeholder Expectations and Requests

```
Stakeholder Input
        ↓
User/Job Stories
        ↓
Documented Requirements (Jira)
```

Stakeholder inputs are gathered and converted into documented Stakeholder requirements
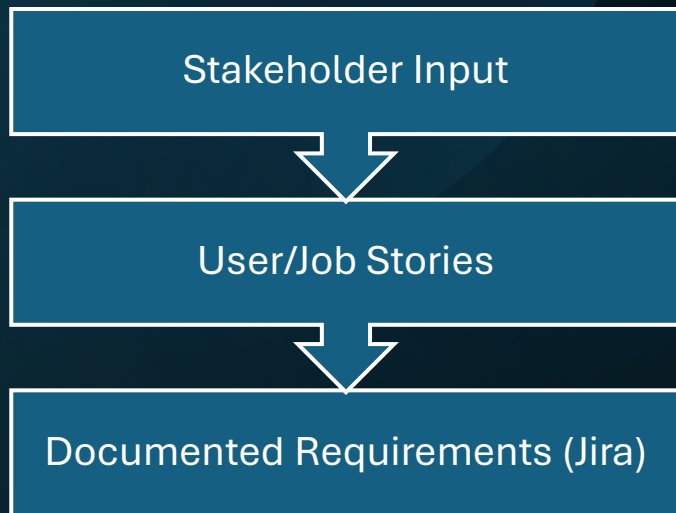
SYS.1.BP1: Obtain stakeholder expectations and requests. Obtain and define stakeholder expectations and requests through direct solicitation of stakeholder input, and through review of stakeholder business proposals (where relevant) and other documents containing inputs to stakeholder requirements, and consideration of the target operating and hardware environment.

Note 1: Documenting the stakeholder, or the source of a stakeholder requirement, supports stakeholder requirements agreement and change analysis (see BP3 and BP4).

1. Direct Engagement with Stakeholders

- Elicit requirements directly from stakeholders through interviews, surveys, and workshops.
- User stories (e.g., "As a user, I want to...") help capture functional needs.
- Job stories focus on tasks (e.g., "When X happens, I want to Y, so that Z").

2. Documenting Requirements

- Convert stakeholder inputs into formal requirements using user and job stories.
- Ensure that all stakeholder requests are documented clearly and thoroughly.
- Requirements should be recorded in a repository like Jira or Confluence.

3. Considering Constraints

- Consider operational, hardware, and environmental constraints while gathering requirements.
- Ensure that the gathered requirements are feasible for the system's scope.
- Align requirements with both business goals and technical capabilities.

User Stories and Job Stories capture different types of requirements

BP1 Examples: Obtaining Stakeholder Expectations

1. Example 1: User Story

- "As a **customer**, I want **to receive email notifications** to **track my order**."
- User story captures the need for real-time order tracking.
- Ensure functional requirements are captured for system notification features.

2. Example 2: Job Story

- "When my **order is shipped**, I want to **be notified via SMS** so that I **know when it will arrive**."
- Job story captures the specific task and goal for real-time communication.
- Ensure task-oriented actions are linked to stakeholder expectations.

3. Best Practices for Eliciting Stories

- Engage stakeholders in workshops to define stories.
- Clarify vague requirements through follow-up questions.
- Capture both technical and non-technical requirements for completeness.

User Story:
- [User] wants to [do something] to [achieve goal]"

Job Story:
- When [situation], I want to [task], so that [desired outcome]"

User Story	Job Story
As a user (User), I want to **receive email notifications (do something) for order** updates (achieve goal)	When my order is shipped (situation), I want to receive an SMS notification (task) so I know when it will arrive (desired outcome)
As a customer (User), I want to log in with a password (do something) to access my account (achieve goal)	When I log in (situation), I want multi-factor authentication (task) to ensure my account is secure (desired outcome)
As a buyer (User), I want to search for products (do something) easily (achieve goal)	When I visit the website (situation), I want to see a search bar at the top (task) so I can quickly find products (desired outcome)
As an admin (User), I want to export user data (do something) for analysis (achieve goal)	When a report is generated (situation), I want to download it in CSV format (task) so I can analyze user trends (desired outcome)
As a user (User), I want to sort items by price (do something) to find cheaper options (achieve goal)	When I view the product list (situation), I want the option to sort by price (task) so I can find cheaper options (desired outcome)
As a customer (User), I want to see product reviews (do something) before making a purchase (achieve goal)	When I look at a product (situation), I want to read reviews (task) so I can decide if it's worth buying (desired outcome)
As a user (User), I want to reset my password (do something) easily (achieve goal)	When I forget my password (situation), I want a reset link emailed to me (task) so I can quickly regain access (desired outcome)
As a guest (User), I want to browse without logging in (do something) to explore products (achieve goal)	When I visit the site (situation), I want to browse as a guest (task) so I don't need to create an account (desired outcome)
As a user (User), I want a detailed order summary (do something) to review items before paying (achieve goal)	When I check out (situation), I want to see a detailed summary of my order (task) so I can review items before paying (desired outcome)
As a shopper (User), I want to filter products by category (do something) to find relevant items faster (achieve goal)	When browsing (situation), I want to filter products by category (task) so I can find relevant items faster (desired outcome)

BP1 Examples: Obtaining Stakeholder Expectations

Using Atlassian Tools for Stakeholder Expectations

Stakeholder Request

↓

Document as User Story in Jira

↓

Track Progress in Dashboard

Stakeholder expectations are captured as user stories in Jira and tracked through project dashboards

1. Jira for User Stories

- Create user stories in Jira by using standard templates.
- Track stakeholder requests and link them to related requirements.
- Use Jira's custom fields to capture priority, business impact, and feasibility.

2. Confluence for Documentation

- Use Confluence to document meeting notes and workshop results.
- Integrate Jira issues (user stories) directly into Confluence pages for easy reference.
- Maintain a single repository for all elicitation documents and ensure traceability.

3. Jira Filters and Dashboards

- Create filters to track the progress of elicited stakeholder requirements.
- Use dashboards to display the status of all user stories and job stories.
- Share dashboards with stakeholders to keep them updated on requirement progress.

BP2: Understand Stakeholder Expectations

Stakeholder Requirements

| Development Team | Testing Team | Management Team |

Consistent understanding across teams helps ensure accurate implementation of stakeholder requirements

> **SYS.1.BP2: Understand stakeholder expectations**. Ensure that all affected parties understand each requirement in the same way.
>
> *Note 2: Examples of affected parties are customers, suppliers, design partners, joint venture partners, or outsourcing parties.*

1. Align Understanding Across Teams

- Ensure all teams (development, testing, management) interpret user and job stories in the same way.
- Regular alignment meetings prevent misinterpretation.
- Avoid conflicting interpretations by discussing expectations early on.

2. Consistency in Documentation

- Use shared documentation to maintain a single source of truth for requirements.
- Ensure that requirements are version-controlled in tools like Confluence.
- Provide regular updates to all parties on any changes or clarifications.

3. Review with Affected Parties

- Hold review sessions with stakeholders to ensure their expectations are understood.
- Verify that all requirements are feasible and clear.
- Obtain confirmation from stakeholders that their needs are accurately captured.

BP2 Examples: Understanding Stakeholder Expectations

Original Requirement ⟩ Clarified Requirement ⟩

Reviewing and clarifying requirements can prevent misinterpretations.

1. Example 1: Cross-Team Review Session

- Hold a meeting with the development, testing, and design teams to review user stories.
- Align on terminology and clarify ambiguous requirements.
- Example: Clarify that "secure login" means multi-factor authentication.

2. Example 2: Stakeholder Confirmation Session
- Present the documented requirements back to stakeholders for confirmation.
- Ensure all parties agree on the wording and meaning of the requirements.
- Example: Confirm that "easy-to-use interface" means a simplified UI design.

3. Best Practices for Consistency
- Use standardized templates for documenting requirements.Ensure all teams have access to the same version of requirements in Confluence.Set up regular review meetings to avoid misunderstandings..

Original User Story / Job Story	Clarified User Story / Job Story
As a user, I want to **reset my password easily** to regain access.	**As a user**, I want to **receive a reset link via email** so that I can securely update my password.
As a customer, I want to **track my order status** in real time to know when it will arrive.	**As a customer**, I want to **receive SMS and email notifications** so I can be updated with every shipping change.
As an admin, I want to **download user data reports** for analysis purposes.	**As an admin**, I want to **export reports in CSV and PDF formats** so I can analyze user activity over 90 days.
When placing an order, I want to **receive a confirmation email** so I know it was processed.	**When placing an order**, I want to **receive confirmation with order details, delivery estimate, and tracking number** so I can stay informed.
As a guest, I want to **browse products without creating an account** so I can explore items before purchasing.	**As a guest**, I want to **filter products and add to a temporary cart** so I can shop without needing to register.
As a buyer, I want to **filter search results by price** so I can easily find cheaper options.	**As a buyer**, I want to **set minimum and maximum price filters** so I can narrow down my search to match my budget.
When logging in, I want to **use multi-factor authentication** to protect my account from unauthorized access.	**When logging in**, I want to **receive one-time passwords via SMS or email** so that my account is securely protected.
As a user, I want to **view product reviews** before purchasing so I can make informed decisions.	**As a user**, I want to **sort reviews by most recent or helpful** so that I can find the most relevant feedback.
As a customer, I want to **cancel my order easily** if it hasn't been shipped yet.	**As a customer**, I want to **cancel my order within 24 hours and receive an automatic refund** so that I can change my mind.
When contacting support, I want to **receive a ticket number** so I can track my support request's progress.	**When contacting support**, I want to **track my ticket status in real time** so I can monitor updates and resolutions.

BP2 Examples: Understanding Stakeholder Expectations

Atlassian Tools for Understanding Stakeholder Expectations

Confluence Document (Stakeholder Expectations)

↓

Team Review

↓

Jira Issue Linking for Tracking

The approval process ensures that all stakeholder needs are formalized and agreed upon before development

1. Confluence for Collaboration

- Use Confluence to host documents for collaborative reviews and feedback.
- Stakeholders can provide feedback directly in the Confluence page comments.
- Keep all teams aligned on the same set of requirements.

2. Jira Issue Linking

- Link related Jira issues to show dependencies between user stories.
- Ensure that all linked requirements are updated consistently across teams.
- Helps teams track interrelated tasks and updates easily.

3. Version Control and Tracking

- Use standardized templates for documenting requirements.
- Ensure all teams have access to the same version of requirements in Confluence.
- Set up regular review meetings to avoid misunderstandings..

BP3:
Agree on Requirements

```
User Stories
      ↓
Stakeholder Review
      ↓
Formalized Requirements
(Jira/Confluence)
      ↓
Sign-Off
```

The approval process ensures that all stakeholder needs are formalized and agreed upon before development

SYS.1.BP3: Agree on requirements. Formalize the stakeholder's expectations and requests into requirements and obtain an explicit agreement from all affected parties to work on these requirements.

Note 3: The agreed stakeholder requirements may be based on feasibility studies and/or cost and schedule impact analysis.

1. Formalizing User and Job Stories

- Convert user and job stories into formal system requirements.
- Each requirement shall be signed off by stakeholders before proceeding.
- Document formalized requirements in Jira or Confluence for tracking.

2. Agreement Process

- All affected parties must review and sign off on the agreed requirements.
- Use version control to manage and track changes to the requirements.
- Agreement helps prevent scope changes during development.

3. Change Management Considerations

- Define a process for managing requirement changes after agreement.
- Use Jira workflows to track approved changes and their impact.
- Communicate changes to all teams involved to avoid misalignment.

BP3 Examples: Agreeing on Requirements

User Story / Job Story

↓

Clarification with Stakeholder

↓

Final Agreement
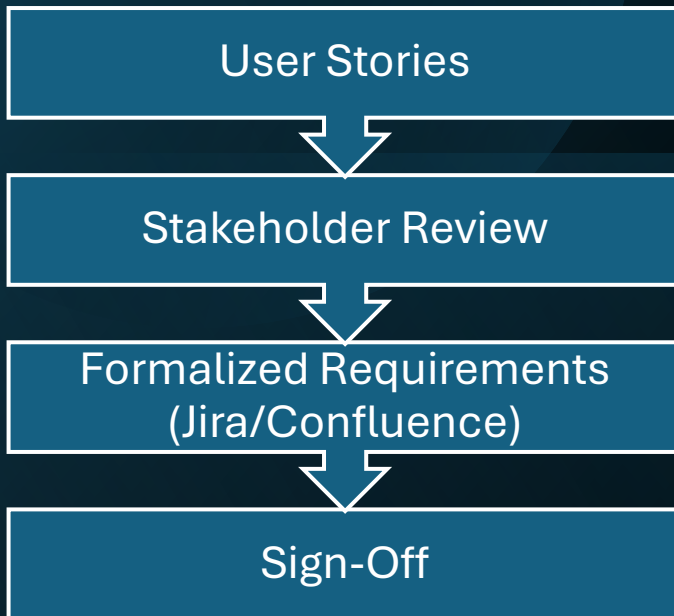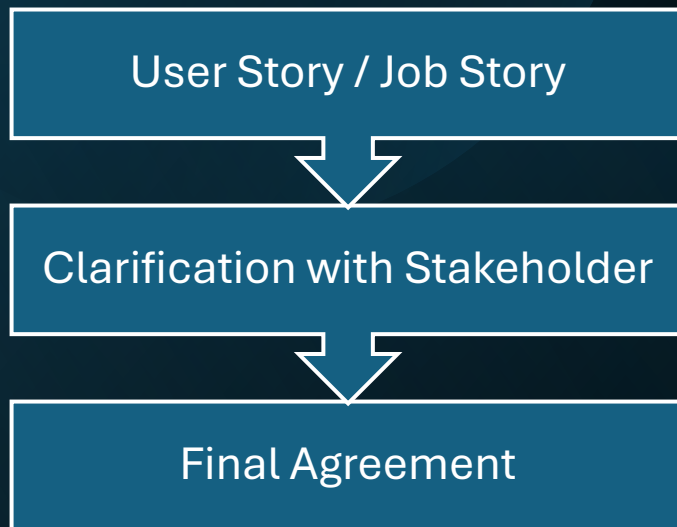
User Stories and Job Stories are clarified with stakeholders and confirmed for development

1. Example 1: Agreement on User Story

- **User Story**: "As a customer, I want to receive real-time notifications for order updates, so I can track my delivery."
- **Clarification**: Agreed upon with the customer to include notifications via both SMS and email.
- **Process**: User story is confirmed with stakeholders and linked to development tasks.

2. Example 2: Agreement on Job Story

- **Job Story:** "When a customer places an order, I want to send a confirmation email, so that they know their order was processed."
- **Clarification:** Stakeholders agree to include additional details like order number and estimated delivery time in the confirmation email.
- **Process:** Job story is finalized and shared with development teams.

3. Best Practices for Agreement

- Ensure User Stories and Job Stories remain consistent with stakeholder language.
- Use Jira to track feedback and sign-off on each story.
- Regularly revisit stories with stakeholders to verify that no scope changes have occurred.

Original User Story / Job Story	Clarification with Stakeholder	Final Agreement
User Story: As a user, I want to reset my password to regain access.	Stakeholder agrees that a password reset link will be sent via email with security checks before updating.	**As a user**, I want to receive a password reset link via email with identity verification, so I can securely regain access.
User Story: As a customer, I want to track my order status in real time.	Agreed that notifications will be sent via SMS and email for each shipping update.	**As a customer**, I want to receive real-time order status updates via SMS and email so I can track my delivery.
Job Story: When placing an order, I want to receive a confirmation email, so I know my order was processed.	Stakeholders agree that confirmation will include order details and estimated delivery.	**When placing an order**, I want to receive a confirmation email with order details and delivery estimate so I know it was processed.
User Story: As a buyer, I want to search for products easily using filters.	Agreed that search filters will include price, category, and rating.	**As a buyer**, I want to search for products using filters like price, category, and rating so I can quickly find what I need.
Job Story: When a user logs in, I want them to use multi-factor authentication for security.	Stakeholders agree to use both SMS and email for delivering one-time passwords.	**When logging in**, I want to use multi-factor authentication with one-time passwords via SMS or email to secure my account.
User Story: As a guest, I want to browse products without creating an account.	Agreed that guest users can add products to a temporary cart without registering.	**As a guest**, I want to browse products and add them to a temporary cart without creating an account so I can shop easily.
Job Story: When users place an order, I want to generate an invoice automatically.	Agreed that invoices will be sent via email and stored in user accounts for registered users.	**When placing an order**, I want an invoice to be generated and sent via email, and stored in my account for future reference.

BP3 Examples: Agreeing on Requirements

Atlassian Tools for Agreement on Requirements

Requirement Documented in Jira

↓

Pending Approval

↓

Approved

↓

Moved to Development
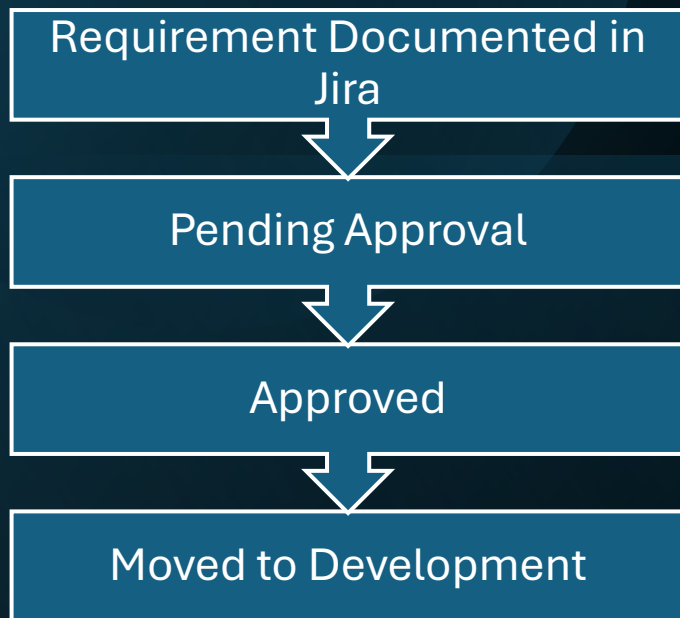
Approval process for stakeholder requirements using Jira

1. Jira Workflows for Sign-Off

- Use Jira workflows to manage the approval process for each stakeholder requirement.
- Create custom workflows for approval stages, including "Pending Approval," "Approved," and "Rejected."
- Track the status of each requirement and obtain formal sign-offs using Jira.

2. Confluence for Documentation and Agreement

- Store finalized agreements and approvals on Confluence pages for easy reference.
- Use Confluence's "Page Approval" plugin to manage sign-offs directly in documentation.
- Maintain an audit trail of who approved each requirement and when.

3. Tracking Approved Requirements

- Use Jira dashboards to track the status of approved, pending, and rejected requirements.
- Share these dashboards with stakeholders to ensure they are updated on the agreement process.
- Ensure only approved requirements move to the development stage.

BP4:
Analyze Stakeholder Requirements Changes

User Story Change

↓

Risk Analysis

↓

Mitigation Strategy
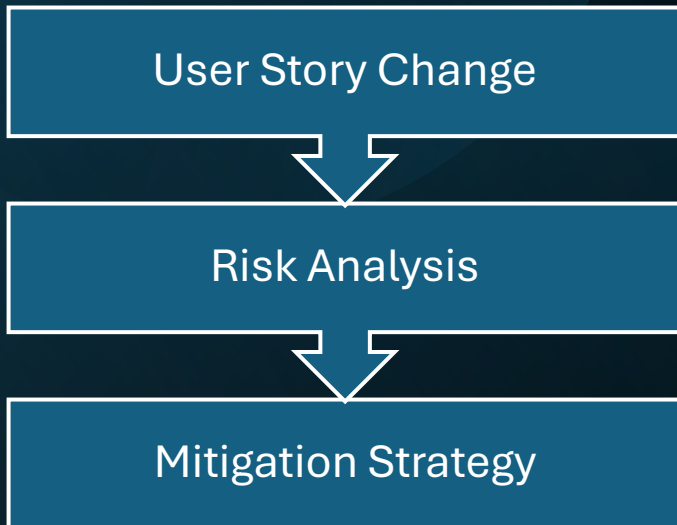
Changes to stakeholder requirements are analyzed for impact and managed to minimize disruptions

SYS.1.BP4: Analyze stakeholder requirements changes. Analyze and manage all changes made to the stakeholder requirements against the agreed stakeholder requirements. Assess the impact and risks and initiate appropriate change control and mitigation actions.

Note 4: Requirements changes may arise from different sources as for instance changing technology, stakeholder needs, or legal constraints.

Note 5: Refer to SUP.10 Change Request Management, if required.

1. Change Analysis Process

- Track and assess all proposed changes to stakeholder requirements .
- Use tools like Jira to monitor the status of changes and their associated risks.
- Ensure that affected stakeholders are notified of changes and potential impacts.

2. Risk and Impact Assessment

- Evaluate how changes could affect project progress or user experience.
- Prioritize changes based on their impact on functionality and stakeholder satisfaction.
- Address potential risks such as increased complexity or user confusion.

3. Stakeholder Communication

- Keep stakeholders informed of changes to stakeholder requirements .
- Ensure that changes are approved by stakeholders before implementation.
- Use Confluence or Jira for documenting and tracking changes.

BP4 Examples : Analyzing Stakeholder Requirement Changes

Original Story

↓

Change

↓

Risk Identified

↓

Mitigation Strategy

Changes are analyzed for risk and impact mitigation strategies are developed to address these risks
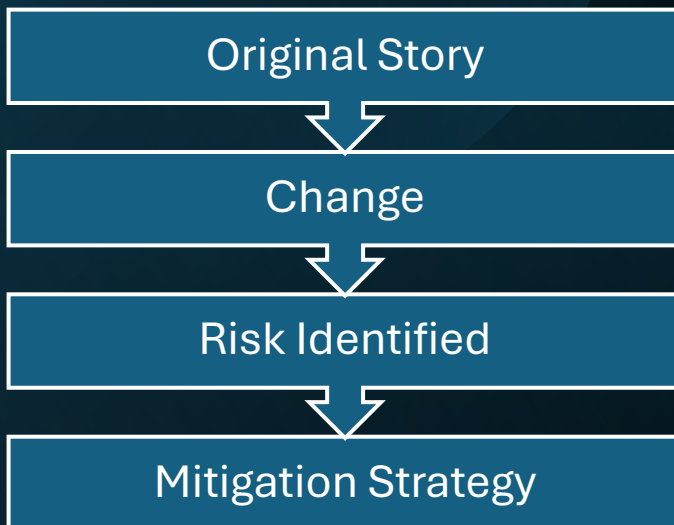
1. Example 1: Change to User Story

- **Original:** "As a user, I want to log in with a password."
- **Change:** "As a user, I want to use two-factor authentication (2FA) for added security."
- **Risk:** Increased complexity for users and additional development time.
- **Mitigation:** Provide clear instructions for using 2FA and offer both SMS and email options.

2. Example 2: Change to Job Story

- **Original:** "When I place an order, I want to receive a confirmation email."
- **Change:** "I want to receive both SMS and email confirmations for order updates."
- **Risk:** Higher notification costs and increased workload for the development team.
- **Mitigation:** Limit SMS notifications to key order updates and optimize notification frequency.

**3. Best Practices for Risk Mitigation**

- Always evaluate changes for their potential risks to timeline, cost, and user experience.
- Use Jira to track the status of changes and the corresponding mitigation strategies.
- Communicate risks and mitigations to all stakeholders before proceeding.

Original Story	Change Requested	Risk Identified	Impact Level	Mitigation Strategy
User Story: As a user, I want to reset my password to regain access.	Add two-factor authentication (2FA) to password reset process.	Increased complexity and user frustration due to extra security step.	Medium	Provide clear instructions for 2FA and offer an option to skip for less sensitive accounts.
Job Story: When placing an order, I want to receive a confirmation email.	Include more details in the confirmation email (order number, delivery date).	Risk of overwhelming users with too much information in a single email.	Low	Simplify email layout, highlighting key details first with additional information easily accessible.
User Story: As a customer, I want to track my order status in real time.	Change from email-only notifications to both SMS and email notifications.	Additional development effort to integrate SMS notifications, increased costs.	Medium	Limit SMS notifications to key updates to reduce costs and complexity.
Job Story: When a user logs in, I want them to use multi-factor authentication for security.	Switch to app-based authentication (instead of SMS).	Users unfamiliar with app-based authentication may struggle, leading to login issues.	High	Provide detailed user guides on setting up and using app-based authentication.
User Story: As a guest, I want to browse products without creating an account.	Allow guest checkout instead of requiring account creation for purchasing.	Loss of customer data for future marketing and order tracking.	High	Provide incentives (e.g., discounts) for users to create accounts after purchase.
Job Story: When a customer cancels an order, I want an automatic refund issued.	Add a 24-hour window for customers to change their mind before refund is processed.	Increased complexity in refund system, potential for user confusion.	Medium	Clearly communicate the 24-hour window during checkout and in the confirmation email.
User Story: As a customer, I want to leave product reviews after making a purchase.	Require reviews to be verified by email confirmation.	Risk of fewer reviews due to extra verification step, reducing feedback volume.	Low	Offer incentives (e.g., discounts) for verified reviews to encourage participation.

BP4 Examples : Analyzing Stakeholder Requirement Changes

Atlassian Tools for Requirement Changes
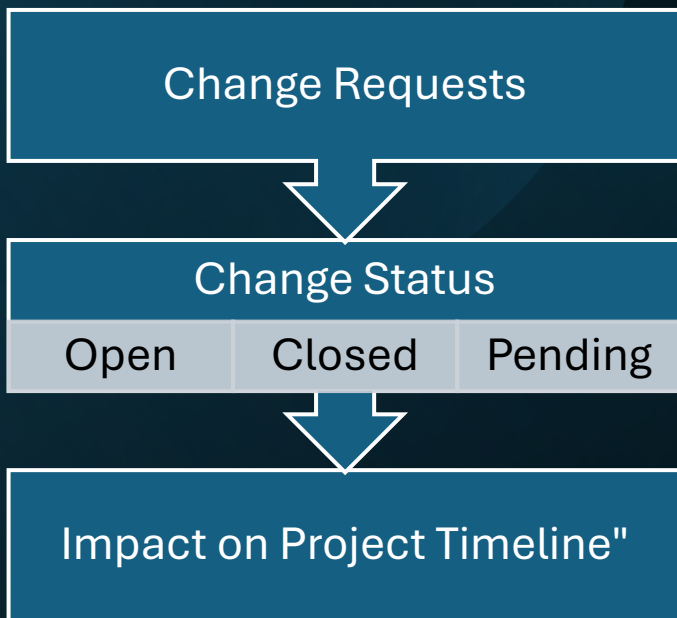
Tracks the status of change requests and their impact on project timelines, ensuring transparency and accountability

Change Requests

Change Status

| Open | Closed | Pending |

Impact on Project Timeline"

1. Jira for Change Requests

- Use Jira's "Change Request" feature to track changes in User Stories and Job Stories.
- Create workflows to assess the impact of each change before approval.
- Link change requests to original stories for full traceability.

2. Tracking Changes in Confluence

- Use Confluence to document impact analysis and decisions related to requirement changes.
- Store all change logs in a dedicated space for transparency.
- Ensure all changes are linked to corresponding Jira issues for consistency.

3. Impact Dashboards

- Use Jira dashboards to show the progress and impact of changes on the project timeline and resources.
- Share dashboards with stakeholders for real-time updates on requirement changes.
- Track risks associated with changes and the status of mitigation strategies.

Change Request ID	User/Job Story	Status	Priority	Impact	Due Date
CR-001	As a user, I want to reset my password	In Progress	High	Increased complexity	15-Oct-24
CR-002	When placing an order, send SMS updates	Approved	Medium	Additional cost	10-Oct-24
CR-003	As a customer, I want to filter by price	In Review	Low	Minimal impact	20-Oct-24
CR-004	When logging in, require multi-factor auth	Rejected	High	Security risk	N/A
CR-005	As a guest, allow browsing without account	Closed	Medium	Improved UX	Completed
CR-006	When placing an order, I want to save my payment method	In Review	High	Increased security measures	18-Oct-24
CR-007	As a buyer, I want real-time inventory tracking	In Progress	Medium	Increased system load	22-Oct-24
CR-008	As a user, I want to receive personalized product recommendations	Pending Review	Low	Privacy concerns	25-Oct-24

Change Request Dashboard Example

BP5: Communicate Requirements Status

Requirement Status

- Agreed
- Pending
- In Development

Visibility into the status of stakeholder requirements is helping ensure transparency and alignment

SYS.1.BP5: Communicate requirements status. Ensure all affected parties can be aware of the status and disposition of their requirements including changes and can communicate necessary information and data.

1. Regular Status Updates 📢

- Use Jira or Confluence to provide stakeholders with regular updates on their requirements.
- Ensure stakeholders are notified of changes or progress.
- Keep communication open to avoid misunderstandings.

2. Track Requirement Disposition ✅

- Communicate whether a requirement is agreed upon, pending, or rejected.
- Use Jira dashboards to show real-time progress on stakeholder requirements.
- Regular updates prevent surprises and maintain alignment with project goals.

3. Communicate Changes Effectively 💬

- Ensure that all changes are clearly communicated to stakeholders in a timely manner.
- Document changes in Confluence and notify stakeholders of updates.
- Hold regular meetings to discuss requirement status and any necessary adjustments.

BP5 Examples : Communicating Requirements Status

Requirement Status

↓

Communication Method

↓

Stakeholder Update
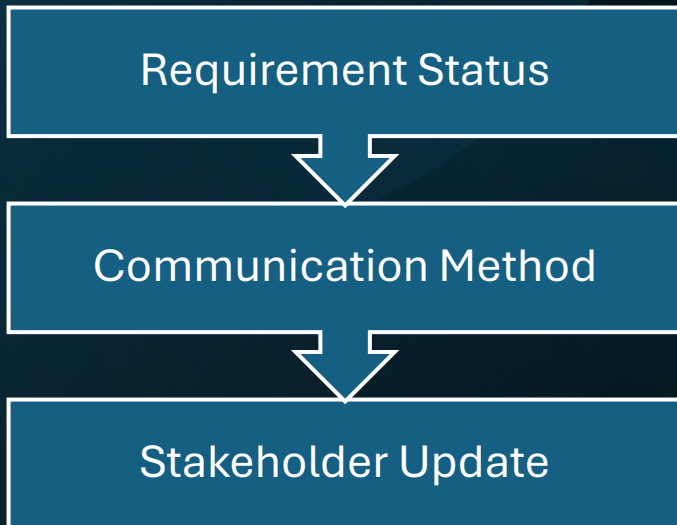
Requirement status is communicated to stakeholders using dashboards and reports

1. Example 1: Real-Time Dashboard

- Use Jira to create a dashboard that shows the real-time status of User Stories and Job Stories.
- Stakeholders can see which stories are in progress, agreed, or pending.
- Example: Order tracking stories updated in real-time.

2. Example 2: Regular Status Reports

- Send weekly status reports to stakeholders outlining requirement progress and changes.
- Include which requirements have been approved and which are awaiting feedback.
- Example: Changes to order confirmation stories communicated clearly.

3. Best Practices for Status Communication

- Use automated notifications to inform stakeholders of any updates to their requirements.
- Ensure all stakeholders have access to real-time updates via Jira dashboards.
- Schedule regular status meetings to provide updates and discuss questions or concerns.

Requirement	Status	Communication Method	Stakeholder Update
As a user, I want to reset my password	In Development	Jira Dashboard	"Password reset functionality is currently being implemented. Expected completion: 15-Oct-2024."
When placing an order, I want to receive a confirmation email	Pending Approval	Weekly Status Report via Email	"Awaiting stakeholder approval for additional order details in confirmation email."
As a customer, I want to track my order status in real time	Approved	Confluence Document with Jira Link	"Real-time tracking notifications via SMS and email have been approved for development."
As a guest, I want to browse products without creating an account	Completed	Email Notification	"Guest browsing feature has been successfully implemented and is now available."
When logging in, I want to use multi-factor authentication	In Review	Jira Comment Thread	"Stakeholder feedback requested on whether to use SMS or app-based authentication."
As a customer, I want to filter products by price	Rejected	Jira Dashboard with Rejection Reasons	"Price filter request rejected due to performance concerns. Feedback available in Jira."
When checking out, I want multiple payment options	Approved	Jira Board with Email Notification	"Multiple payment options have been approved for development and are scheduled for implementation."
As a customer, I want to leave product reviews	Completed	Email Notification & Confluence Summary	"Product review feature is live. Stakeholders can see details in Confluence."

Change Request Dashboard Example

Atlassian Tools for Requirements Status Communication

Requirement Status

- Agreed
- Pending
- Rejected

Dashboard provides stakeholders with real-time updates on the status of their requirements

1. ira for Real-Time Status

- Use Jira to show the real-time status of stakeholder requirements (agreed, pending, or rejected).
- Create Jira filters to track the progress of specific User Stories and Job Stories.
- Share Jira boards with stakeholders so they can track the status directly.

2. Confluence for Reports and Updates

- Use Confluence to generate weekly reports on requirement status, including recent changes and approvals.
- Set up automated email notifications to stakeholders whenever a change in status occurs.
- Provide a centralized Confluence space where all requirement statuses are documented and accessible.

3. Dashboards for Transparency

- Create dashboards in Jira to provide stakeholders with an overview of all their requirements, including progress and changes.
- Use Confluence dashboards to display visual reports of requirement status.
- Ensure stakeholders can see all relevant updates in real-time.

Challenges in Requirements Elicitation

Major challenges in requirements elicitation and provides key areas to focus on for overcoming them

1. Stakeholder Communication Gaps 💬

- Misunderstandings between stakeholders and development teams can lead to unclear or incomplete requirements.
- Different terminology and expectations may create confusion.
- Regular workshops and feedback loops can help bridge communication gaps.

2. Difficulty in Prioritizing Requirements

- Stakeholders may struggle to prioritize their needs, leading to conflicting priorities.
- Misalignment between business goals and user needs may result in delays.
- Use structured techniques (e.g., MoSCoW) to prioritize requirements effectively.

3. Handling Requirement Changes

- Frequent changes to User Stories and Job Stories can disrupt project timelines and increase complexity.
- Uncontrolled changes can lead to scope creep and misaligned expectations.
- Implementing change management practices with Jira can help track and manage changes systematically.
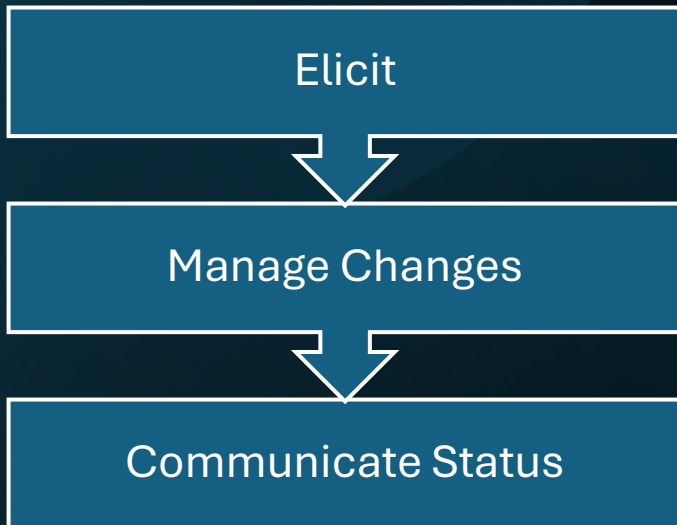
Stakeholder Communication Gaps

Difficulty in Prioritizing

Handling Requirement Changes

Summary and Q&A

Elicit

↓

Manage Changes

↓

Communicate Status

System requirements, traceability, and communication are essential for project success

1. Eliciting Stakeholder Needs Effectively

- Gathering clear and concise User Stories and Job Stories ensures alignment between stakeholders and the project team.
- Use structured techniques like interviews and workshops to capture all necessary inputs.
- Document and refine requirements collaboratively with stakeholders to avoid misunderstandings.

2. Managing Requirement Changes

- Analyze the impact of changes to User Stories and Job Stories on the project timeline and stakeholder expectations.
- Use tools like Jira and Confluence to track, assess, and communicate changes.
- Properly managing changes avoids scope creep and ensures stakeholder needs are met.

3. Communicating Requirement Status

- Keep stakeholders updated on the status of their requirements through regular communication channels such as Jira dashboards, email notifications, and Confluence updates.
- Regular status reports and meetings help prevent miscommunication and ensure transparency throughout the project.
- Communicating progress and challenges early ensures stakeholder alignment and mitigates risks.