Our Way of Working: A Team Guidebook

**Introduction**

**Objective:**

* Establish a unified framework for efficient work management, collaboration, and communication across our geographically dispersed team, enabling us to achieve our automation goals and deliver high-quality solutions.

**Current Status:**

* Currently, we are experiencing challenges with work visibility, prioritization, and balanced workload distribution across our India and America teams. Additionally, there are communication gaps and coordination issues, particularly around incident and problem management, that impact our overall efficiency and productivity.

**Future View:**

* By adopting the principles and practices in this guidebook, we envision a future where our team operates seamlessly across geographical boundaries. We will have clear roles and responsibilities, streamlined processes, effective communication, and a shared understanding of our work, enabling us to deliver exceptional results and achieve our automation vision.

Tabla de contenido

[1. Jira Management 3](#_Toc176268036)

[2. Operational Roles and Responsibilities 6](#_Toc176268037)

[3. Service Management 9](#_Toc176268038)

# 1. Jira Management

**Objective:**

* Efficiently manage and track all work, including project tasks, documentation efforts, incident management, and problem resolution, using Jira as the central platform.
* Ensure clear visibility, prioritization, and balanced workload distribution across geographically dispersed teams in India and America.
* Facilitate effective collaboration, communication, and decision-making through transparent Jira practices.

**Key Principles:**

* **All Work in Jira:** Every piece of work, regardless of its nature, must be captured in a Jira ticket.
* **Clear and Descriptive Tickets:** Tickets should have clear titles, detailed descriptions, acceptance criteria, assignees, priorities, due dates, and story point estimates.
* **Regular Updates and Closure:** Ticket statuses should be updated regularly, and completed tickets should be closed promptly.
* **Collaborative Backlog Grooming:** Regular grooming sessions should be held to refine, prioritize, and estimate upcoming work, ensuring story points are assigned or reviewed.
* **Adapting Jira for Non-Development Work:** Utilize Kanban boards or other suitable project management methodologies to visualize and manage non-development work. Define clear workflows and track progress using Jira boards and reports.
* **Separate Tracking for Operational Work:** Estimate and track JSM tickets separately using story points and a dedicated swimlane or board. Allocate capacity for JSM work during sprint planning.
* **Prioritization and Integration:** Prioritize JSM tickets based on SLAs or other criteria. Integrate high-priority JSM tickets into sprints alongside project stories.
* **Team Collaboration and Communication:** Encourage regular sync-ups, maintain a shared knowledge base, and promote cross-training to enhance collaboration and communication between team members handling JSM tickets and project work.
* **Automation and Proactive Measures:** Automate routine tasks and implement proactive problem management to reduce the number of incoming JSM tickets and free up team members' time for more complex work.

**Specific Guidelines:**

* **Jira Ticket Creation:**
  + **Mandatory for All Work:** All work, including project tasks, documentation, incident reports, and problem-solving activities, must be captured in Jira tickets.
  + **Clear and Comprehensive Information:** Ensure tickets include:
    - Clear and concise titles.
    - Detailed descriptions outlining the scope, requirements, and expected outcomes.
    - Assignee(s) responsible for the work.
    - Realistic due dates.
    - Priorities based on urgency and impact.
    - Story point estimates using the Fibonacci scale, reflecting the team's shared understanding of effort and complexity.
* **Jira Ticket Management:**
  + **Regular Updates:** Update ticket statuses regularly to reflect progress.
  + **Timely Closure:** Close completed tickets promptly.
  + **Clear Communication:** Use comments to document discussions, decisions, and any roadblocks encountered.
* **Jira Project/Work Management:**
  + **Backlog Grooming:** Conduct regular backlog grooming sessions to refine, prioritize, and estimate upcoming work, including documentation tasks and incident/problem resolution. Ensure story points are assigned or reviewed during grooming.
  + **Sprint Planning:** Conduct sprint planning sessions to commit to a set of work for the upcoming sprint, considering both project stories and high-priority JSM tickets.
  + **Sprint Execution and Monitoring:** Monitor sprint progress and make adjustments as needed. Track both project work and JSM ticket resolution.
  + **Sprint Retrospectives:** Conduct sprint retrospectives to identify areas for improvement in both project and operational work management.
* **Jira for Operational Work (JSM):**
  + **Estimate and Track Separately:** Assign story points to JSM tickets and track their progress on a separate swimlane or board.
  + **Allocate Capacity:** Dedicate a portion of the team's capacity for handling JSM tickets during sprint planning.
  + **Prioritize and Integrate:** Prioritize JSM tickets based on SLAs or other criteria and integrate high-priority tickets into sprints.
  + **Limit Work in Progress:** Set WIP limits for both JSM tickets and project stories to ensure a balanced workload.
* **Jira Reporting and Metrics:**
  + **Utilize Dashboards:** Leverage Jira's reporting and dashboard capabilities to track team performance, identify bottlenecks, and measure progress toward goals.
  + **Track Key Metrics:** Monitor metrics such as:
    - Documentation completion rates.
    - Incident response times.
    - Problem resolution rates.
    - Story point completion and velocity for both project work and JSM tickets.
  + **Regular Reviews:** Review and analyze metrics regularly to identify areas for improvement and inform decision-making.
* **Addressing Geographical Challenges:**
  + **Clear Ticket Ownership:** Establish clear guidelines for ticket assignments and ownership across geographically dispersed teams.
  + **Transparent Communication:** Ensure open and proactive communication on ticket updates and progress across time zones.
  + **Foster Collaboration:** Encourage cross-team collaboration and knowledge sharing, especially for complex documentation projects or problem-solving efforts.
  + **Consistent Story Point Estimation:** Ensure consistent understanding and application of story point estimation across both teams.
* **Leveraging Epics:**
  + **Project Organization:** Utilize Epics to define and organize projects, providing clear project boundaries and fostering cross-team collaboration.
  + **Workload Distribution:** Assign Epics to specific teams or individuals to ensure balanced workloads and strategic resource allocation across regions.
  + **Communication and Alignment:** Use Epics as a centralized hub for project-related information and foster cross-team alignment on project objectives, timelines, and dependencies.
* **Team Structure Considerations:**
  + **Single Scrum Master:** Leverage the Scrum Master's expertise to facilitate Agile processes, guide story point estimation, and coordinate activities across both regions.
  + **Distributed Teams:** Adapt practices to accommodate the challenges of having team members spread across India and America, ensuring clear communication and collaboration.

# 2. Operational Roles and Responsibilities

**Objective:**

* Clearly define the roles within the team, incorporating the distribution of incident and problem management responsibilities, to ensure clarity, accountability, and effective collaboration.
* Facilitate seamless coordination and knowledge sharing across geographically dispersed teams in India and America, particularly in areas with specialized skills like scripting and networking.

**Key Roles and Responsibilities (Adapting to Skill Distribution and Shared Responsibilities):**

* **Senior Architects/Architects:**
  + Design and develop the overall architecture for automation solutions, leveraging their expertise and experience.
  + Define technical standards and guidelines for the team to follow.
  + Collaborate with other team members to ensure alignment with business requirements and technical feasibility.
  + Provide technical guidance and mentorship to less experienced team members.
  + Review and approve technical designs and documentation.
  + **Shared Responsibility - Incident Management:**
    - Participate in incident triage and resolution, leveraging their technical expertise to identify root causes and provide solutions.
    - Contribute to post-incident reviews and identify areas for improvement in system design or automation to prevent future incidents.
  + **Shared Responsibility - Problem Management:**
    - Analyze incident trends and contribute to identifying underlying problems.
    - Participate in root cause analysis and propose solutions to prevent problems from recurring.
    - Collaborate with other team members to implement preventive measures.
* **Senior Engineers/Engineers:**
  + Develop, test, and deploy automation scripts and workflows, applying their technical skills to implement solutions.
  + Troubleshoot and resolve issues with automation solutions.
  + Monitor and maintain the performance and reliability of automation systems.
  + Collaborate with architects to ensure alignment with architectural designs and technical standards.
  + Document automation solutions and processes.
  + **Specialized Skills (Scripting):**
    - The few team members with scripting expertise will take ownership of script development, sharing their knowledge and providing guidance to others as needed.
  + **Specialized Skills (Networking):**
    - Network engineers will take charge of the physical networking aspects, ensuring smooth integration with automation solutions.
  + **Shared Responsibility - Incident Management:**
    - Actively participate in incident resolution, leveraging their technical skills and knowledge of the systems.
    - Document incident details and resolution steps in Jira Service Management.
    - Contribute to post-incident reviews and identify areas for improvement in automation or processes to prevent future incidents.
  + **Shared Responsibility - Problem Management:**
    - Report and escalate recurring incidents or patterns that suggest underlying problems.
    - Participate in problem-solving efforts and contribute to root cause analysis.
    - Assist in implementing solutions and preventive measures.
* **Scrum Master:**
  + Facilitate Agile processes for both India and America teams, adapting to the team's skill distribution and project requirements.
  + Guide and coach team members on Agile principles and practices.
  + Facilitate sprint planning, daily stand-ups, sprint reviews, and retrospectives, ensuring effective participation from all team members.
  + Remove impediments and foster a collaborative team environment that encourages knowledge sharing and skill development.
  + Ensure consistent understanding and application of story point estimation across both teams.
  + Coordinate activities and communication between geographically dispersed teams, paying special attention to areas with specialized skills.
  + **Oversight of Incident and Problem Management:**
    - Ensure incidents and problems are tracked and addressed effectively in Jira Service Management.
    - Facilitate communication and collaboration between team members involved in incident and problem resolution.
    - Monitor incident and problem trends to identify areas for improvement in processes or automation.

**Additional Considerations:**

* **Skill-Based Task Assignment:** Assign tasks based on individual skills and expertise, ensuring efficient utilization of resources and fostering a sense of ownership.
* **Knowledge Sharing:** Encourage knowledge sharing and mentorship, particularly in areas with limited expertise, to build team capabilities and reduce dependency on specific individuals.
* **Flexibility and Adaptability:** Be prepared to adjust roles and responsibilities as needed based on evolving team needs, project requirements, and skill development.
* **Open Communication:** Foster a culture of open communication and transparency, allowing team members to raise concerns, seek help, and share knowledge freely.
* **Collective Ownership:** Emphasize collective ownership of incident and problem management, ensuring everyone contributes to maintaining system stability and reliability.

# 3. Service Management

**Objective:**

* Establish clear and efficient processes for managing incidents, problems, and changes to ensure system stability, minimize disruptions, and facilitate timely resolution.
* Foster effective communication and collaboration among team members and stakeholders throughout the service management lifecycle.
* Leverage Jira Service Management to track, prioritize, and resolve incidents and problems effectively.

**Incident Management:**

* **Definition:** An incident is any unplanned interruption or reduction in the quality of an IT service or configuration item that affects its operation.
* **Process:**
  1. **Incident Identification and Logging:** Incidents can be reported by users, monitoring systems, or other sources. All incidents must be logged in Jira Service Management with relevant details, including:
     + Description of the issue
     + Impact and urgency
     + Affected systems or components
     + Reporter's contact information
  2. **Incident Triage and Prioritization:** Incidents are triaged based on their impact and urgency, using predefined criteria and service level agreements (SLAs).
  3. **Incident Assignment and Investigation:** Incidents are assigned to appropriate team members based on their skills and expertise. The assignee investigates the incident, gathers additional information, and attempts to resolve it.
  4. **Incident Resolution and Closure:** Once the incident is resolved, the assignee documents the resolution steps and closes the ticket in Jira Service Management.
  5. **Post-Incident Review:** For major incidents, a post-incident review is conducted to identify root causes, contributing factors, and areas for improvement.
* **Communication:**
  1. **Stakeholder Communication:** Keep stakeholders informed about the incident status, progress, and estimated resolution time through regular updates and notifications.
  2. **Internal Communication:** Facilitate effective communication and collaboration among team members involved in incident resolution, leveraging Jira Service Management comments and other communication channels.
* **Escalation:**
  1. **Escalation Procedures:** Define clear escalation procedures for incidents that cannot be resolved within the defined SLAs or require additional expertise or resources.
  2. **Escalation Communication:** Ensure timely and effective communication during escalations, keeping all relevant stakeholders informed.

**Problem Management:**

* **Definition:** A problem is the underlying cause of one or more incidents. Problem management aims to identify and resolve the root cause of problems to prevent future incidents.
* **Process:**
  1. **Problem Identification:** Problems can be identified through incident trends, recurring incidents, or proactive analysis.
  2. **Problem Logging:** Problems are logged in Jira Service Management with relevant details, including:
     + Description of the problem
     + Related incidents
     + Potential impact
     + Assignee responsible for investigation
  3. **Root Cause Analysis:** The assignee conducts a thorough root cause analysis to identify the underlying cause of the problem.
  4. **Solution Development and Implementation:** The team develops and implements a solution to address the root cause and prevent the problem from recurring.
  5. **Problem Closure:** Once the solution is implemented and verified, the problem is closed in Jira Service Management.
* **Collaboration:**
  1. **Cross-functional Collaboration:** Foster collaboration between different teams and roles to identify and resolve problems effectively.
  2. **Knowledge Sharing:** Encourage knowledge sharing and documentation of problem resolutions to prevent similar issues in the future.

**Change Management:**

* **Definition:** Change management is the process of controlling and managing changes to IT services and configuration items to minimize disruptions and ensure successful implementation.
* **Process:**
  1. **Change Request:** A change request is submitted in Jira Service Management, outlining the proposed change, its impact, and the required approvals.
  2. **Change Assessment:** The change is assessed for its potential impact, risks, and benefits.
  3. **Change Approval:** The change request is reviewed and approved by relevant stakeholders.
  4. **Change Implementation:** The change is implemented according to the approved plan.
  5. **Change Review:** The change is reviewed to ensure it was implemented successfully and met its objectives.
* **Communication:**
  1. **Stakeholder Communication:** Keep stakeholders informed about upcoming changes, their potential impact, and the implementation timeline.
  2. **Change Advisory Board (CAB):** Establish a CAB to review and approve significant changes.

**Jira Service Management:**

* **Incident and Problem Tracking:** Utilize Jira Service Management to log, track, and manage incidents and problems, ensuring visibility and accountability.
* **Workflows and Automation:** Customize workflows and leverage automation to streamline incident and problem management processes.
* **Reporting and Metrics:** Utilize Jira Service Management reports and dashboards to track key metrics, such as incident resolution times, problem resolution rates, and change success rates.

**Additional Considerations:**

* **Service Level Agreements (SLAs):** Define SLAs for different types of incidents and problems to ensure timely resolution and manage stakeholder expectations.
* **Knowledge Base:** Create and maintain a knowledge base to document common issues, solutions, and troubleshooting steps.
* **Continuous Improvement:** Regularly review and improve service management processes based on feedback, metrics, and lessons learned.

**4. Meeting and Scheduling**

**Objective:**

* Establish clear guidelines and best practices for conducting effective meetings, fostering collaboration, and ensuring efficient communication across our geographically dispersed team.
* Address scheduling considerations, agendas, expected outcomes, and tools for virtual meetings to maximize productivity and engagement.
* Implement a hybrid approach for daily stand-ups, combining written updates with optional Teams calls, to accommodate time zone differences and promote inclusivity.

**Meeting Types and Frequency:**

* **Daily Stand-ups (Hybrid Approach):**
  + **Purpose:** Provide a brief overview of daily progress, identify any roadblocks, and facilitate quick coordination within teams.
  + **Frequency:** Daily written updates, with optional Teams calls on Monday, Wednesday, and Friday.
  + **Format:**
    - **Written Updates:** Each team member provides a concise update in Jira or Confluence before their workday starts, including:
      * What they accomplished yesterday
      * What they plan to do today
      * Any blockers or challenges they are facing
    - **Teams Calls (Optional):** Scheduled at a convenient time for both India and America teams, these calls are used for:
      * Discussing blockers or challenges in more detail
      * Brainstorming solutions collaboratively
      * Fostering team connection and communication
      * Calls will be recorded for those who cannot attend live.
* **Sprint Planning Meetings:**
  + **Purpose:** Plan the work for the upcoming sprint, including selecting and estimating user stories, defining sprint goals, and assigning tasks.
  + **Frequency:** At the beginning of each sprint.
  + **Duration:** 1-2 hours, depending on the sprint length and complexity of the work.
  + **Attendees:** Scrum Master, Product Owner, and the entire team.
* **Sprint Review Meetings:**
  + **Purpose:** Review the work completed during the sprint, demonstrate new features or functionality, and gather feedback from stakeholders.
  + **Frequency:** At the end of each sprint.
  + **Duration:** 1-2 hours, depending on the sprint length and amount of work completed.
  + **Attendees:** Scrum Master, Product Owner, team members, and relevant stakeholders.
* **Sprint Retrospective Meetings:**
  + **Purpose:** Reflect on the sprint, identify areas for improvement, and develop action plans to implement changes in the next sprint.
  + **Frequency:** At the end of each sprint.
  + **Duration:** 1 hour.
  + **Attendees:** Scrum Master and the entire team.
* **Other Meetings:**
  + **Team Meetings:** Regular team meetings can be held to discuss broader topics, share updates, and foster team building.
  + **Project Meetings:** Project-specific meetings can be scheduled as needed to discuss project progress, address specific challenges, or make decisions.
  + **One-on-One Meetings:** Regular one-on-one meetings between managers and team members can be used for performance feedback, career development discussions, and addressing individual concerns.

**Scheduling Considerations:**

* **Time Zones:** Carefully consider the time difference between India and America when scheduling meetings. Aim for times that are convenient for the majority of attendees, rotating meeting times if necessary to ensure fairness.
* **Meeting Invitations:** Send meeting invitations well in advance, including the agenda, expected outcomes, and any required preparation.
* **Meeting Reminders:** Send reminders before the meeting to ensure attendees are prepared and aware of the meeting time.
* **Flexibility:** Be flexible and accommodating to individual needs and preferences, allowing for remote participation or alternative meeting times when possible.

**Agenda and Expected Outcomes:**

* **Clear Agenda:** Prepare a clear agenda for each meeting, outlining the topics to be discussed and the desired outcomes.
* **Time Management:** Allocate time for each agenda item and stick to the schedule.
* **Action Items:** Clearly define action items and assign ownership to ensure follow-through after the meeting.
* **Meeting Notes:** Take detailed meeting notes and share them with all attendees and relevant stakeholders.

**Virtual Meeting Tools and Best Practices:**

* **Microsoft Teams:** Utilize Microsoft Teams for video conferencing, screen sharing, and chat functionality.
* **Audio and Video Quality:** Ensure good audio and video quality for all participants.
* **Engagement:** Encourage active participation from all attendees, using interactive features like polls, Q&A, and whiteboarding.
* **Recording:** Record meetings for those who cannot attend or for future reference.
* **Accessibility:** Provide accommodations for team members with disabilities or limited internet connectivity.

**Additional Considerations:**

* **Meeting Etiquette:** Foster a culture of respect and inclusivity during meetings. Encourage active listening, avoid interruptions, and ensure everyone has an opportunity to contribute.
* **Meeting Effectiveness:** Regularly evaluate the effectiveness of meetings and make adjustments as needed to improve productivity and engagement.
* **Communication Channels:** Utilize appropriate communication channels for different types of communication, such as email, chat, or project management tools.

**Important Considerations for the Hybrid Stand-up Approach**

* **Clear Expectations:**
  + Set clear expectations for the written updates, including the level of detail required and the deadline for submission
* **Review and Follow-up:**
  + Ensure that someone (e.g., the Scrum Master or team lead) reviews the written updates daily and follows up on any blockers or concerns
* **Culture of Communication:**
  + Foster a culture of open communication and encourage team members to reach out for help or clarification whenever needed, regardless of the channel used
* **Evaluate and Adapt:**
  + Regularly assess the effectiveness of this hybrid approach and make adjustments as needed based on team feedback and evolving needs

**5. Team Norms**

**Objective:**

* Establish shared values, expectations, and guidelines for communication, collaboration, and conduct within the team to foster a positive, inclusive, and high-performing work environment.
* Promote a culture of trust, respect, and mutual support across geographically dispersed teams in India and America.

**Communication:**

* **Open and Transparent Communication:**
  + Encourage open and honest communication among all team members.
  + Create a safe space for sharing ideas, concerns, and feedback without fear of judgment or reprisal.
  + Proactively communicate updates, challenges, and any potential impacts on project timelines or deliverables.
  + Utilize appropriate communication channels (e.g., Microsoft Teams, email, Confluence) based on the urgency and nature of the message.
* **Active Listening:**
  + Practice active listening during conversations, demonstrating genuine interest and understanding towards others' perspectives.
  + Ask clarifying questions and paraphrase to ensure accurate comprehension.
  + Avoid interrupting or formulating responses while others are speaking.
* **Respectful Communication:**
  + Communicate with respect and courtesy, valuing diverse perspectives and backgrounds.
  + Avoid any language or behavior that could be perceived as offensive, discriminatory, or exclusionary.
  + Be mindful of cultural differences and adapt communication styles to ensure effective understanding and collaboration across teams.
* **Constructive Feedback:**
  + Provide constructive feedback in a timely and respectful manner, focusing on specific behaviors and outcomes rather than personal attacks.
  + Offer suggestions for improvement and acknowledge areas of strength.
  + Be open to receiving feedback and use it as an opportunity for growth and development.
* **Timely Response:**
  + Respond to messages and requests in a timely manner, acknowledging receipt and providing updates on progress or estimated response time.
  + Set realistic expectations for response times and communicate any delays or changes in availability.
* **Clarity and Conciseness:**
  + Strive for clarity and conciseness in written and verbal communication, avoiding jargon and unnecessary complexity.
  + Use clear and simple language that is easily understood by all team members, regardless of their technical background or expertise.
  + Organize information in a logical and structured manner to facilitate comprehension.
* **Cross-Cultural Sensitivity:**
  + Be mindful of cultural differences in communication styles, values, and expectations.
  + Adapt your communication approach to ensure effective understanding and collaboration across teams in India and America.
  + Seek to learn about and appreciate different cultural perspectives, fostering an inclusive and respectful team environment.

**Collaboration:**

* **Shared Goals and Vision:**
  + Align team members around shared goals and a common vision for the project or initiative.
  + Ensure everyone understands the overarching objectives and how their individual contributions fit into the bigger picture.
  + Regularly revisit and reinforce the team's goals and vision to maintain focus and motivation.
* **Collective Ownership:**
  + Foster a sense of collective ownership and responsibility for the team's success.
  + Encourage everyone to take initiative, contribute their best efforts, and support each other in achieving shared goals.
  + Celebrate successes as a team and learn from setbacks together.
* **Knowledge Sharing:**
  + Actively share knowledge, skills, and experiences with other team members, promoting continuous learning and development.
  + Document processes, procedures, and lessons learned in Confluence or other shared repositories.
  + Organize regular brown bag sessions or knowledge-sharing workshops.
  + Encourage mentorship relationships between experienced and less experienced team members.
* **Mentorship and Support:**
  + Provide mentorship and guidance to less experienced team members, fostering a culture of growth and development.
  + Offer constructive feedback, share insights, and provide opportunities for learning and skill enhancement.
  + Create a supportive environment where team members feel comfortable asking questions and seeking help.
* **Collaboration Tools:**
  + Utilize collaboration tools like Confluence, Jira, and Microsoft Teams to facilitate seamless communication and information sharing across teams.
  + Leverage these tools to document project plans, track progress, share updates, and collaborate on documents and deliverables.
* **Cross-Team Collaboration:**
  + Encourage collaboration and knowledge exchange between teams in India and America.
  + Leverage opportunities for joint projects, shared learning sessions, and virtual team-building activities to foster stronger connections and understanding.
  + Celebrate diversity and leverage the unique perspectives and strengths of each team to achieve collective success.

**Conduct:**

* **Professionalism:**
  + Maintain a high level of professionalism in all interactions, both within the team and with external stakeholders.
  + Adhere to company policies and ethical guidelines.
  + Represent the team and the organization in a positive and respectful manner.
* **Punctuality:**
  + Be punctual for meetings and deadlines, respecting others' time and commitments.
  + Communicate proactively if you anticipate any delays or conflicts.
* **Accountability:**
  + Take ownership of your work and be accountable for your actions and deliverables.
  + Meet commitments and deadlines, or communicate proactively if adjustments are needed.
  + Learn from mistakes and take corrective action to prevent future issues.
* **Positive Attitude:**
  + Maintain a positive and solution-oriented attitude, even when faced with challenges or setbacks.
  + Focus on opportunities for growth and learning, rather than dwelling on negativity.
  + Encourage and support your teammates, fostering a positive and collaborative team spirit.
* **Conflict Resolution:**
  + Address conflicts in a constructive and respectful manner, seeking mutually agreeable solutions.
  + Focus on open communication, active listening, and understanding different perspectives.
  + Escalate conflicts to a manager or team lead if necessary to facilitate resolution.
* **Work-Life Balance:**
  + Encourage a healthy work-life balance, promoting self-care and well-being among team members.
  + Respect individual boundaries and encourage time off for rest and rejuvenation.
  + Foster a supportive environment where team members feel comfortable discussing any challenges they may be facing in balancing work and personal life.

**Specific Team Norms:**

* **Story Point Estimation:**
  + **Consistent Understanding:** Ensure a shared understanding of story point estimation across both India and America teams.
  + **Collaborative Estimation:** Use collaborative estimation techniques, such as Planning Poker, to reach consensus on story points.
  + **Regular Calibration:** Regularly review and calibrate story point estimations to ensure consistency and accuracy.
* **Knowledge Sharing:**
  + **Documentation:** Document processes, procedures, and lessons learned in Confluence or other shared repositories.
  + **Brown Bag Sessions:** Organize regular brown bag sessions for team members to share knowledge and expertise on specific topics.
  + **Mentorship:** Encourage mentorship relationships between experienced and less experienced team members.
* **Collective Ownership of Incident and Problem Management:**
  + **Shared Responsibility:** Everyone on the team is responsible for contributing to incident and problem resolution, regardless of their primary role.
  + **Proactive Approach:** Encourage a proactive approach to identifying and addressing potential issues before they escalate into incidents or problems.
  + **Continuous Improvement:** Continuously seek ways to improve incident and problem management processes, leveraging automation and knowledge sharing.
* **Meeting Attendance:**
  + **Importance of Attendance:** Attending scheduled meetings is crucial for effective communication, collaboration, and decision-making.
  + **Timely Arrival:** Be punctual for meetings and come prepared to contribute.
  + **Active Participation:** Engage actively during meetings, share your insights, ask questions, and provide feedback.
  + **Hybrid Stand-up Attendance:**
    - **Written Updates: Mandatory:** Provide daily written stand-up updates in Jira or Confluence, regardless of your ability to attend the Teams calls.
    - **Teams Calls: Encouraged:** While attendance is optional, make an effort to join the Teams calls whenever possible to foster real-time interaction and collaboration.
  + **Sprint Planning and Review Attendance:** Active participation in sprint planning and review meetings is mandatory to ensure alignment on sprint goals, commitment to deliverables, and effective feedback loops.
* **Work Tracking and Updates:**
  + **Timely Updates:** Regularly update Jira tickets and other relevant project management tools to reflect progress, challenges, and any changes in scope or timelines.
  + **Accurate Story Point Estimation:** Ensure consistent and accurate estimation of story points for all tasks, facilitating effective planning and workload management.
  + **Transparent Communication:** Proactively communicate any blockers or delays that may impact project timelines or deliverables.
  + **Regular Reporting:** Participate in regular reporting activities, such as sprint reviews and retrospectives, to share progress and identify areas for improvement.

6**. Documentation Standards**

**Objective:**

* Establish clear guidelines and templates for creating high-quality, consistent, and comprehensive documentation that facilitates knowledge sharing, collaboration, and efficient onboarding of new team members.
* Ensure documentation is easily accessible, maintainable, and up-to-date, reflecting the latest changes and best practices.

**Documentation Types and Scope:**

* **Architectural Designs:**
  + Document the overall architecture of automation solutions, including system diagrams, component interactions, and data flows.
  + Use standard architectural notations and diagrams (e.g., UML, ArchiMate) to visualize the design.
  + Clearly define the purpose, scope, and constraints of the architecture.
  + Include relevant decision points and rationale behind design choices.
* **Process Flows:**
  + Document the step-by-step procedures for various processes, such as incident management, problem management, change management, and automation workflows.
  + Use flowcharts or other visual representations to illustrate the sequence of steps and decision points.
  + Clearly define roles and responsibilities for each step in the process.
  + Include any relevant input/output data and decision criteria.
* **Automation Scripts:**
  + Document the purpose, functionality, and usage of automation scripts.
  + Include clear comments and explanations within the script code to enhance readability and maintainability.
  + Provide instructions on how to execute and troubleshoot the scripts.
  + Document any dependencies or prerequisites for running the scripts.
* **Knowledge Base Articles:**
  + Create knowledge base articles to capture solutions to common issues, troubleshooting steps, and frequently asked questions.
  + Organize articles in a logical and searchable manner using tags and categories.
  + Keep articles up-to-date and relevant, reflecting the latest changes and best practices.
* **Other Relevant Documentation:**
  + Meeting notes, project plans, technical specifications, user guides, and any other documentation relevant to the team's work.

**Documentation Standards:**

* **Clarity and Conciseness:** Use clear and concise language, avoiding jargon and unnecessary complexity.
* **Completeness:** Ensure documentation is comprehensive and covers all relevant aspects of the topic.
* **Accuracy:** Ensure information is accurate, up-to-date, and reflects the latest changes and best practices.
* **Consistency:** Maintain consistency in formatting, style, and terminology across all documentation.
* **Accessibility:** Make documentation easily accessible to all team members through a centralized repository like Confluence.
* **Version Control:** Use version control systems (e.g., GitLab, Bitbucket) to track changes and maintain a history of documentation revisions.
* **Review and Approval:** Implement a review and approval process for critical documentation to ensure quality and accuracy.

**Templates and Guidelines:**

* **Provide templates for common documentation types:**
  + Architectural Design Template
  + Process Flow Template
  + Automation Script Template
  + Knowledge Base Article Template
* **Develop guidelines for documentation style, formatting, and terminology:**
  + Writing Style Guide
  + Documentation Formatting Guide
  + Glossary of Terms

**Tools and Platforms:**

* **Confluence:** Utilize Confluence as the central repository for all team documentation, ensuring easy access and collaboration.
* **GitLab/Bitbucket:** Use version control systems to manage and track changes to documentation, particularly for automation scripts and code-related documents.
* **Diagramming Tools:** Leverage diagramming tools (e.g., Lucidchart, draw.io) to create visual representations of architectural designs and process flows.

**Responsibilities:**

* **All Team Members:**
  + Create and maintain documentation related to their areas of expertise.
  + Adhere to documentation standards and guidelines.
  + Participate in the review and approval process.
* **Architects:**
  + Take ownership of architectural design documentation.
  + Ensure alignment with technical standards and guidelines.
* **Automation Engineers:**
  + Document automation scripts and workflows thoroughly.
  + Keep documentation up-to-date with any changes or enhancements.
* **Scrum Master:**
  + Facilitate the documentation process and ensure adherence to standards.
  + Encourage knowledge sharing and documentation best practices within the team.

**7. Automation Framework**

**Objective:**

* Define a structured and standardized approach to automation, encompassing the tools, technologies, and best practices for developing, testing, deploying, and maintaining automation scripts and workflows.
* Ensure consistency, reliability, and maintainability of automation solutions across the team.
* Foster collaboration and knowledge sharing in automation development and implementation.

**Automation Approach:**

* **Tool Selection:**
  + Identify and select appropriate automation tools and technologies based on project requirements, team expertise, and available resources.
  + Consider factors such as ease of use, scalability, integration capabilities, and community support.
  + Examples of potential tools include Ansible, Terraform, Python, PowerShell, Jenkins, GitLab CI/CD, etc.
* **Coding Standards and Best Practices:**
  + Establish and adhere to coding standards and best practices to ensure code quality, readability, and maintainability.
  + Use meaningful variable names, comments, and consistent formatting.
  + Implement error handling and logging mechanisms for effective troubleshooting.
  + Follow modular and reusable code design principles.
* **Version Control:**
  + Utilize GitLab or Bitbucket for version control of automation scripts and related code.
  + Follow branching strategies and code review processes to ensure code quality and collaboration.
  + Maintain a clear and organized repository structure for easy navigation and access.
* **Testing and Validation:**
  + Implement comprehensive testing and validation procedures to ensure the accuracy and reliability of automation solutions.
  + Utilize unit tests, integration tests, and end-to-end tests to cover different scenarios and edge cases.
  + Automate testing processes wherever possible to improve efficiency and reduce manual effort.
* **Deployment and Release Management:**
  + Establish a well-defined deployment and release management process to ensure smooth and controlled rollout of automation solutions.
  + Utilize CI/CD pipelines (e.g., GitLab CI/CD, Jenkins) to automate build, test, and deployment processes.
  + Implement proper versioning and rollback mechanisms to manage potential issues during deployment.
* **Monitoring and Maintenance:**
  + Implement monitoring and logging mechanisms to track the performance and health of automation systems.
  + Proactively identify and address potential issues before they impact operations.
  + Regularly review and update automation scripts and workflows to reflect changes in requirements or infrastructure.

**Collaboration and Knowledge Sharing:**

* **Code Reviews:** Conduct regular code reviews to ensure adherence to coding standards, identify potential improvements, and share knowledge among team members.
* **Pair Programming:** Encourage pair programming to facilitate collaboration, knowledge transfer, and improve code quality.
* **Shared Libraries and Modules:** Develop and maintain shared libraries and modules to promote code reusability and reduce redundancy.
* **Documentation:** Document automation solutions, including their purpose, functionality, dependencies, and usage instructions, following the established documentation standards.
* **Training and Workshops:** Organize training sessions and workshops to enhance team members' automation skills and knowledge.

**Responsibilities:**

* **All Team Members:**
  + Contribute to the development, testing, and maintenance of automation solutions based on their skills and expertise.
  + Adhere to coding standards and best practices.
  + Participate in code reviews and knowledge-sharing activities.
* **Architects:**
  + Provide technical guidance and oversight on automation architecture and design.
  + Ensure alignment with overall technical standards and guidelines.
* **Automation Engineers:**
  + Take ownership of developing, testing, and deploying automation scripts and workflows.
  + Collaborate with architects and other team members to ensure successful implementation.
* **Scrum Master:**
  + Facilitate collaboration and communication within the team regarding automation efforts.
  + Ensure automation tasks are integrated into the overall project plan and tracked effectively in Jira.

**By establishing and adhering to this automation framework, we can streamline processes, improve efficiency, and reduce manual effort, ultimately enabling us to achieve our automation goals and deliver high-quality solutions.**

**Conclusion**

This "Our Way of Working" guidebook serves as a foundation for establishing a collaborative, efficient, and high-performing team environment. By embracing the guidelines and best practices outlined in this document, we can overcome geographical barriers, leverage individual strengths, and achieve collective success in our automation endeavors.

We encourage all team members to actively participate in shaping and evolving our way of working, fostering a culture of continuous improvement and innovation. Let's work together to build a stronger, more connected, and more successful team!