**BALANE, Daphne P.**

**Iteration Planning Meeting (IPM)**

An Agile iteration cycle starts with an iteration planning meeting where the team arranges tasks for the upcoming period. Led by the product owner, the team discusses and assigns tasks until they agree on a goal.

To achieve the meeting's goal within a set timeframe, an iteration planning meeting agenda is used. This agenda covers iteration goals, team input, and task distribution. The complete agenda includes:

* Attendees: List of participants and their roles.
* Capacity: Each team member's availability to determine capacity.
* Iteration Goals: Developers and product owner detail the initial planning goal and commit to it.
* Relevant Issues: Product owner shares important updates, and team members express concerns.
* Story Analysis and Estimating: Team evaluates user stories, estimating time for completion. Relative sizing might be used.
* Timebox: Meeting time limit, ideally not exceeding four hours for a two-week iteration.
* Velocity: Calculating the team's previous iteration speed.

Agile iteration planning follows a previous iteration's conclusion and retrospective. The software team collaboratively defines the scope, estimating task durations using feedback and historical data.

Before the formal iteration planning, teams estimate story values through story point estimation. The product owner prioritizes Product Backlog Items (PBIs), sets iteration goals, and defines completion criteria. The meeting involves these activities:

* Discussing technical aspects, dependencies, and implementation for the top PBI.
* Establishing acceptance criteria for each story.
* Estimating effort for each story.
* Selecting stories within the team's capacity.
* Breaking down large stories if needed.
* Self-assigning tasks based on skills and capacity.
* Estimating ideal task completion hours.
* Optionally creating an iteration backlog.
* Adjusting tasks if someone exceeds capacity.
* Finalizing iteration goals and committing to them.

Inputs to the iteration planning meeting consist of feedback, issues, objectives, and user stories. Outputs of the meeting include defined scope, task assignments, and agreement on goals. Capacity planning drives Agile iteration planning. The team selects high-priority items, estimates work, and commits to it. Capacity is gauged based on ideal hours, availability, and dedication percentage. More experienced teams can predict capacity better due to established velocity, while newer teams need multiple iterations to improve accuracy.

**Kick-off (Dev kick off with Business Analyst & Quality Analyst)**

Initiating a software development project with a kick-off meeting is a pivotal stage. This meeting engages Business Analysts (BAs), Quality Assurance (QA) experts, and the development team to establish the project's direction. It unifies all stakeholders on aspects such as project scope, requirements, anticipations, and quality benchmarks. Below is a detailed guide on how to execute a kick-off meeting:

1. Preparation:

Prior to the kick-off meeting, the project manager or team leader should make the following arrangements:

* Project Documentation: Compile all pertinent project documents like the project charter, requirement papers, user narratives, wireframes, and related materials.
* Invitations: Dispatch invitations to development team members, Business Analysts, Quality Assurance professionals, and other pertinent stakeholders.
* Agenda: Construct an agenda outlining meeting subjects, objectives, and expected results.

1. Meeting Commencement:

Open the meeting by welcoming attendees and clarifying the kick-off meeting's objective. Emphasize the significance of cooperation, communication, and synchronization for a triumphant project result.

1. Project Overview:

Present an overarching view of the project, involving its aims, extent, and desired results. This lays the groundwork for ensuing discussions.

1. Roles and Responsibilities:

Define the duties of each team member. Highlight the roles of Business Analysts, QA specialists, and developers in ensuring project success.

1. Review of Requirements:

Business Analysts elucidate the detailed project requisites, encompassing user narratives, functional specifications, and design papers. Allow room for inquiries and clarifications from the development team and QA professionals.

1. Technical Dialogue:

The development team explores the technical facets of the project, including architectural choices, technological preferences, integration points, and challenges that require addressing.

1. QA and Testing Overview:

QA specialists provide an overview of the testing approach, discussing testing types (e.g., unit, integration, system), testing environments, data requirements, and quality benchmarks.

1. Acceptance Criteria:

Collaboratively set lucid acceptance criteria for each user story or feature. These criteria define when a feature is deemed finished and fulfills specified requisites.

1. Timeline and Milestones:

Deliberate the project timeline, encompassing pivotal milestones, deliverables, and projected completion dates for distinct phases. Confirm that the timeline aligns with stakeholder anticipations.

1. Communication Strategy:

Lay out the communication management plan for the project. Discuss update frequency, communication channels, and reporting hierarchy.

1. Risk and Issue Exchange:

Identify potential risks and challenges that could influence the project. Encourage candid discussions and brainstorming to devise strategies for managing these issues.

1. Q&A and Open Dialogue:

Allocate time for queries, concerns, and open discussions. This is an opportunity for team members to express thoughts, seek clarifications, and share insights.

1. Action Items and Next Steps:

Sum up crucial takeaways, decisions, and identified action items from the meeting. Assign responsibilities and set deadlines for each task.

1. Closing Comments:

Extend gratitude to participants for their involvement and input. Highlight the value of cooperation and teamwork in achieving project success.

1. Follow-Up:

Post-meeting, distribute meeting notes and action items to all attendees. Ensure timely tracking and resolution of identified action items.

A triumphant development kick-off meeting hinges on effective communication, collaboration, and alignment among Business Analysts, QA specialists, and the development team. It forges the basis for seamless project execution by guaranteeing a shared comprehension of project goals, requisites, and expectations among all stakeholders.

**Desk Check (Dev desk check with Business Analyst & Quality Analyst)**

A desk examination, often termed a "desk review" or "desk walk-through," constitutes a collaborative practice within software development. In this process, developers, Business Analysts (BAs), and Quality Assurance (QA) specialists collaborate to assess and authenticate alterations in code or other project components. The primary aim is to detect and address potential concerns in the initial stages of development before they escalate into more significant complications. Here is an extensive explanation of how to conduct a desk check involving developers, BAs, and QA professionals:

1. Preparation:

Ahead of the desk check, ensure the following are ready:

* Code Modifications: Verify that the code changes or project components set for review are primed for assessment. This could entail code updates, feature incorporations, or other deliverables.
* Invitations: Dispatch invitations to pertinent team members, including developers, BAs, and QA professionals. Specify the date and time for the desk check.
* Agenda: Create an agenda outlining the objectives of the desk check, the elements under review, and the expected results.

1. Meeting Commencement:

Commence the desk check meeting by extending a welcome to all participants and restating the review's intention. Highlight the collaborative nature of the procedure and the significance of early issue identification.

1. Code Assessment:

If the desk check involves code changes, the developer accountable for the modifications presents their code to the team. This may involve walking through the code, elucidating the alterations made, and discussing the reasoning behind them.

1. Validation of Requirements:

Business Analysts assess the code changes against the original requirements and user narratives. They confirm that the executed functionality aligns with the intended features and user expectations.

1. Quality Assurance Evaluation:

QA specialists analyze the code changes from a testing angle. They address the test cases meant for verifying functionality, pinpoint possible areas of concern, and guarantee code compliance with quality benchmarks.

1. Discussion and Clarifications:

Allocate time for queries, clarifications, and open dialogues. Encourage team members to pose questions regarding the code, requisites, design choices, or any other elements necessitating further clarity.

1. Validation of Acceptance Criteria:

Collectively validate that the code changes fulfill the prescribed acceptance criteria. If any criteria are unmet, explore ways to address them before proceeding.

1. Technical Considerations:

Tackle any technical considerations, such as performance ramifications, security worries, and integration points. Verify that the changes are congruent with the broader technical framework.

1. Issue Identification:

If any issues or apprehensions surface during the desk check, discuss them candidly and determine suitable actions to rectify them. This could encompass modifying the code, revisiting requirements, or adjusting the testing strategy.

1. Subsequent Steps:

Recap the pivotal takeaways from the desk check meeting. Define the subsequent steps, which may encompass necessary code amendments, further testing, or extra conversations.

1. Action Items:

Record any action items stemming from the desk check, including assignments for developers, BAs, and QA professionals. Allocate responsibilities and define deadlines for each action item.

1. Post-Meeting Follow-Up:

After the meeting, disseminate meeting notes and action items to all participants. Guarantee the timely tracking and completion of the identified action items.

1. Concluding Comments:

Express appreciation to all participants for their contributions to the desk check. Reinforce the value of cooperation and communication in upholding code quality and realizing project objectives.

A fruitful desk check guarantees the alignment of code changes with requisites, quality standards, and technical considerations. By involving developers, Business Analysts, and QA professionals in the review process, you can identify potential issues early, foster collaboration, and facilitate a smoother development and testing cycle.