**Contents**

[1. Purpose 4](#_Toc103669479)

[2. Deployment approaches 4](#_Toc103669480)

[3. Release management schedule 4](#_Toc103669481)

[4. Service value chain 5](#_Toc103669482)

[5. Contribution to Service value chain 6](#_Toc103669483)

[6. Release and deployment management phases 6](#_Toc103669484)

[7. Release checklist 7](#_Toc103669485)

[8. Requirements traceability matrix 7](#_Toc103669486)

[9. Service Test and pilots 7](#_Toc103669487)

[10. Release acceptance criteria 8](#_Toc103669488)

[11. Service test plan 8](#_Toc103669489)

[12. Service test case 8](#_Toc103669490)

[13. RACI Matrix 9](#_Toc103669491)

[14. Release post implementation review 9](#_Toc103669492)

[15. Feedback on Release Planning 9](#_Toc103669493)

[16. Feedback on release execution 11](#_Toc103669494)

[17. Key performance Indicators 12](#_Toc103669495)

**Document Control**

**Document Version History**

This table shows a record of significant changes to the document.

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description of Change** |
| 0.1 | 27/01/2022 | Swapnil Wale | DRAFT |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Approvals**

This table shows the approvals on this document for circulation, use and withdrawal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Approver** | **Title/Authority** | **Approval Remarks** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## 1. Purpose

The objective of Release and Deployment Management is to organize, schedule, and control the migration of releases from development to test and production environments. The major purpose of this ITIL process is to protect the live environment's integrity and disaster recovery while also releasing the appropriate components.

## 2. Deployment approaches

* Big Bang approaches- The deployment of all new or changed components is done together. This approach is occasionally required when dependencies prevent the use of old and new components.
* Phased approach- The components are deployed one after the other in scheduled phases. This procedure is carried out as many times as necessary until the deployment is complete.
* Pull approach- New or updated software is made available in a controlled repository, and users can choose to download it to their client devices. Users can regulate the schedule of updates, and it can be used with service request management to allow users to request software only when it is required.
* Push approach- Components are deployed to the target systems by pushing them from a central system.
* Manual approach- This approach requires manual approach while deploying
* Automated deployment- The deployment is done in an automated way by meeting some schedules.

## 3. Release management schedule

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Release no** | **Release type** | **Description** | **Approval status** | **Performed by** | **Release frequency** | **Planned started date** | **Planned end date** | **Actual release date** |
| Qw31 | Major release | The new version (#2.7) is intended at upgrading our current SAP version to the S/4 HANA version. | Pending | Sneha | Quarterly | 2/2/22 | 2/25/22 | 2/18/22 |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

## 4. Service value chain

Chart, diagram, funnel chart

Description automatically generated

Service value chain involves a bunch of processes involved in creation of products and services which provides value to customers.

* Plan- The goal of plan value chain activity is to ensure that everyone in the business has a shared understanding of the vision, present status, and improvement direction for all four dimensions, as well as all goods and services.

The four dimensions under plan activity include:

1. Engage- The purpose of this activity is to provide a good understanding of clients and stakeholder needs, expectations and building continual engagement with stakeholders
2. Obtain/Build- The obtain/build value chain activity's goal is to ensure that service components are available when and when they're needed, and that they meet agreed-upon standards.
3. Deliver and Support- The deliver and support value chain activity's goal is to make sure that services are delivered and supported in accordance with agreed-upon specifications and stakeholder expectations.

* Continual Improvement- This practise must be included at all service levels of the business and be a part of the day-to-day activities.

## 5. Contribution to Service value chain

|  |  |
| --- | --- |
| **SVC activities** | **Release deployment activities** |
| Plan | Focuses on release policies, guidelines, and timeframes that are driven by the organization's strategy and service portfolio, where the scope, size, and content of the release are planned and controlled. |
| Improve | This phase focuses on the releases that need to be improved and the lessons learned from prior releases. |
| Engage | The priority is on involving the consumers and users who will benefit from the new release. It also includes engaging with release planners, architects, business analysts, developers etc. |
| Design and transition | **focuses on the development and implementation of new or modified contribution services in a live environment.** |
| Obtain and build | **Emphasis on components required for a certain release and ensuring that they are made available to customers in a controlled manner. It also includes the phases involved from design to deployment.** |

## 6. Release and deployment management phases

* Release planning- A release plan specifies what will be included in a release as well as how it will be deployed into production. Typically, the plan addresses:
* What changes will be made in this release?
* Who will be benefitted or impacted by the release?
* How is release identified?
* What, if any, risk does the release pose?
* The release's target audience (i.e., what users, customers, organisations will be impacted)
* A clear approval chain that specifies which stakeholders can approve the change request at each stage of the release.
* The term "ownership" refers to the group of people who are ultimately responsible for the release.
* Schedule and plan for deployment
* Release building and testing- After a plan has been established and authorised by change management, the responsible teams must build and test the release, which includes both software and documentation, as well as any other elements specified in the release plan.

Documentation is often prepared at the start of this process to ensure that developers can build the release package as precisely and rapidly as possible and accurate records should be kept throughout the build process so that the build process may be repeated if necessary.

* Deployment- Beginning when change management permits the release package to be deployed to the target environments, the release package is deployed to the live environment. Handoff to service operations and early-life support completes the deployment phase.

ITIL recommends doing a lot of planning and preparation ahead of time, such as validating that the target group is ready for the deployment, having an acceptance criterion identifying and attempting to minimise any potential risks or disruptions, and describing how each component of the release will be distributed in order.

* Review and closing a deployment- It's time to examine and learn from the entire process once the release has been delivered. Feedback is received, and performance goals are evaluated, with the outcomes being reviewed and shared by all parties involved. Release notifications are sent to owners, after each release phase.

Reviews should be thorough and meticulous, ensuring that all quality criteria have been met, that sufficient knowledge transfer and training has taken place, and that any known problems, repairs, or changes have been properly documented. A comprehensive Post Implementation Review, or PIR, should also be conducted by change management.

## 7. Release checklist

The list contains items that need to be completed before software gets released

|  |  |  |  |
| --- | --- | --- | --- |
| **Activities** | **ID** | **Responsible** | **Done** |
| Create and install configuration scripts | 53ed | Developer |  |
| Debug and test the code | 13132 | Developer |  |
| Scan virus on all release media | 12896 | Release team |  |
| Release is taken from authorized source | 34231 | Release team |  |

## 8. Requirements traceability matrix

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Requirement description** | **Business need** | **Requested by** | **Department** | **Status** | **Design** | **Test case** |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

## 9. Service Test and pilots

Testing and validation must be done at numerous levels as part of a coordinated effort with Service Validation and Testing.

Pilot testing is a sort of software testing in which a set of users uses the software in its entirety prior to the software's official launch or deployment.

1. Train any people who will be participating.
2. Document any required processes.
3. Constant contact and involvement with clients and users
4. Determining the levels of support required for the actual release
5. Identifying and resolving issues before final deployment
6. Documenting improvements and incorporating them into future

## 10. Release acceptance criteria

The release acceptance criteria should include the following:

* + Is the release working as it should in all development and test environments?
  + Has the back-up plan been put to the test?
  + Is there a training schedule in place, and have all necessary teams been adequately trained?
  + Has the release been subjected to comprehensive User Acceptance Testing (UAT) with participation from all impacted customer and user groups?
  + Is the release performing as intended and defect-free in test environments?
  + Is it possible to deploy the release with a workaround if testing has found any issues, or should the release be postponed?
  + Have all impacted support teams been trained on any new support features introduced because of the release?

## 11. Service test plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Milestone** | **Type of test** | **Test scenarios** | **Number of cycles** | **Hours per cycle** | **Total Estimated hours** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## 12. Service test case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test case** | **Test summary** | **Steps** | **Expected result** | **Actual result** | **Status** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## 13. RACI Matrix

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Activities** | **Release and deployment manager** | **Change Manager** | **Testing Manager** | **Delivery Manager** | **Build Teams** | **Deployment Teams** | **Customer Service Manager** |
| Raise and update release ticket | AR | I | I | I | I |  |  |
| Release planning | AR | CI | CI | CI | CI | CI |  |
| Build the release | A | I | I | I | RCI | CI |  |
| Preparation of updating the documents | AR | RI | RI | RI | R | I | I |
| Release authorization | I | A | I | RI | I | I | CI |
| Deployment of release | A | I | I | I | CI | CR | I |
| Release rollback | AR | R | I | CI | CI | RI | CI |
| Release communication | AR | I | I | I | I | I | I |
| Release communication | AR | I | I | I | I | I | I |
| Release reporting | AR | CI | CI | CI | R | R | CI |

## 14. Release post implementation review

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Project Name: | |  | | Change Number | |  | |
| Release Manager | |  | | Project Number: | |  | |
| Release Date: | |  | | PIR Date: | |  | |
| **#** | **Description** | | **Date** | | **Status** | | **Owner** |
|  |  | |  | |  | |  |
|  |  | |  | |  | |  |
|  |  | |  | |  | |  |
|  |  | |  | |  | |  |

## 15. Feedback on Release Planning

|  |  |  |
| --- | --- | --- |
| **Review #1** | **Review Name: Resource allocation** | **Owner: PMO** |
| **Which types of resources were allocated in the initial plan: Physical/Human/Budget / etc.?** | | |
| **How were the resource risks managed?** | | |
| **How can the allocation process be simplified?** | | |
| **Any extension was need?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #2** | **Review Name: System downtime communicated** | **Owner: Project Manager** |
| **Were the required downtime windows planned in advance or requested ad-hoc?** | | |
| **Was the scope of each downtime achieved, or was an extension required?** | | |
| **How efficient was the process of getting the approvals for each downtime?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #4** | **Review Name: Change Instructions Quality** | **Owner: QA Manager** |
| **How many clarifications were required per each change request?** | | |
| **How close was the actual change to the requested change?** | | |
| **Were the acceptance criteria met?** | | |
| **Who requested the most changes?** | | |
| **Which planned activity required the most changes?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #5** | **Review Name: Processes followed** | **Owner: QA Manager** |
| **Which processes were followed?** | | |
| **Were the processes ITIL / PMBOK approved?** | | |
| **Were any in-house changes made to the processes?** | | |
| **Do the team-members recommend the processes for the next implementations?** | | |
| **Will an addition of a flow chart of the processes be helpful?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #6** | **Act Name: Scope for Improvement** | **Owner: Project Manager** |
| **What were the lessons learned for the next implementation planning?** | | |
| **How close was the actual implementation to the planned one?** | | |
| **Which changes do we need to make to the work plan / Gantt chart for the next implementation?** | | |
| **Were the notifications effective in communicating to stakeholders?** | | |

## 16. Feedback on release execution

|  |  |  |
| --- | --- | --- |
| **Review #1** | **Review Name: Tasks Started on Time** | **Owner: PMO** |
| **Specify what type of tasks started on time (HW / SW / Integration / etc.)** | | |
| **What percentage of the tasks started on time?** | | |
| **How many predecessors did the tasks that started on time have?** | | |
| **What was the average duration of the tasks that started on time?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #2** | **Review Name: Status communication were adequate** | **Owner: Project Manager** |
| **How many follow up questions and clarifications were needed after each communication was sent?** | | |
| **Which format was used (E-mail, face-to-face, video conferencing, etc.), and which platform (Outlook, Trello board, etc.)?** | | |
| **How many people were in the correspondence communications?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #3** | **Review Name: Were all of the stakeholders informed** | **Owner: Project Manager** |
| **How many unplanned meetings were scheduled by a stakeholder in order to fully understand the status of the implementation?** | | |
| **Were all of the stakeholders informed in each status update correspondence?** | | |
| **How many of the changes were approved by all the stakeholders?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #4** | **Review Name: Adequate issue management** | **Owner: Project Manager** |
| **Which percentage of the issues required an escalation to resolve?** | | |
| **What was the average timeframe of resolving the issues?** | | |
| **How many resources were required on average to resolve an issue?** | | |
| **Can release deployment be automated?** | | |

|  |  |  |
| --- | --- | --- |
| **Review #5** | **Review Name: Scope for Improvement** | **Owner: Project Manager** |
| **What were the lessons learned for the release / roll-out?** | | |
| **Do we need a dedicated war-room staffed 24/7 for the duration of the hyper-care period?** | | |
| **Should the processes / methodology / procedures be changed or modified for the next release?** | | |
| **Is it necessary for the resources to physically be present in the release location?** | | |
| **How was third party response in release deployment?** | | |

## 17. Key performance Indicators

|  |  |
| --- | --- |
| **KPIs** | **Description** |
| % Of release success rate | the percentage of the total number of closed releases that were successfully executed in the specific period |
| % Of release defect rate | the percentage of releases that result in incidents in a given time period, |
| Number of known release errors | Number of known release errors category wise over time |
| % of approved releases that don't cause an issue | the percentage of approved releases that do not result in an Incident in a given period |
| Satisfied users | % of satisfied users |