TECH THAT MATTERS

Elevaite365

Change Management Policy & Procedure

Version 1.0

PURPOSE

This policy aims to control planned and unplanned changes within the environment and infrastructure of Elevaite365 (hereby referred to as organization), including cloud services. It ensures that changes to production, development, testing, configuration, and administration of shared assets, services, and processes follow an approved, auditable, and secure procedure.

SCOPE

This policy applies to organization employees, contractors, and operations and should be read with other relevant policies (e.g., Access Control, Incident Management). It covers all forms of change that impact the production, development, testing, and administration of organization IT assets, including (but not limited to) operating systems, software installations and configurations, networking devices, public computing services, storage and database platforms, application development and support, and documentation. Access requests for corporate servers and applications also fall under this policy.

DEFINITION

Below is a breakdown of the different terms used in this document.

- **Change:** An alteration of the form, nature, or content of an IT system, process, or component from its current state to a different state.
- **Standard Change:** A pre-authorized, low-risk change following a well-established, documented procedure typically requires no further approval for each occurrence.
- **Minor Change:** A non-standard, low-impact change that affects only a single user or system and does not significantly alter business operations.
- **Major Change:** A non-standard, high-impact change that affects multiple users, systems, or applications and can pose a significant risk if not carefully managed.
- **Emergency Change:** Minor or significant changes must be implemented urgently to address critical situations like severe outages, security breaches, or high-priority incidents.
- **CR (Change Request):** A formal document or record is used to propose, track, and manage a change. It typically includes essential details like reason, scope, and risk level.
- **Change Requestor:** The individual or team initiating a change by submitting the Change Request for consideration and tracking.
- **Change Reviewer:** The individual or team that examines the details of a Change Request, classifies its type, and routes it to the appropriate approver(s).
- **Change Approver:** The individual or group authorized to review the proposed change’s merits and decide whether to approve, reject, or postpone it.
- **Change Implementer:** The person or team responsible for the approved change according to the prescribed method, timeline, and scope.
- **Change Manager:** The role or individual overseeing the change process ensures that all changes follow formal guidelines, approvals, and documentation.
- **Rollback Plan:** A predefined set of actions or procedures to revert a system, application, or environment to its original state if the implemented change fails or causes issues.
- **Post-Implementation Review (PIR):** After a change is deployed, a structured review is performed to confirm whether it met its objectives, evaluate success criteria, and capture lessons for future improvements.

RESPONSIBILITIES

1. **Department Heads**

    a. They have primary responsibility for implementing this policy within their respective areas.

2. **Information Security Group (ISG)**
    a. Implements this policy under leadership guidance.
    b. Coordinates with Department Heads to ensure all changes adhere to security and compliance requirements.

POLICY

1. **Change Management Lifecycle**
    a. The organization shall initiate, implement, authorize, and review all changes consistently.
    b. Duties, roles, and responsibilities must be segregated between development/test environment(s) and production environment(s).
    c. Access controls must prevent unauthorized changes to production systems.
2. **Testing and Authorization**
    a. All changes require sufficient testing and documentation before moving to production.
    b. Impact assessments must consider security, operational risk, and potential downtime.
    c. Management signoff, authorization, or peer review is required before implementation.
3. **Documentation and Rollback**
    a. Change documentation must include impact analysis, testing results, and roll-back procedures.
    b. Every change must have a defined roll-back or contingency plan to be activated if issues arise in production.
4. **Compliance**
    a. All changes must comply with information security standards, best practices, and organizational or client requirements.
    b. Test data and accounts must be removed or disabled before changing into production.

PROCEDURE

Change Initiation

1. **Role:** Change Requestor
2. **Description:**
    a. A user, team member, or system monitoring identifies a need for change.
    b. The requester classifies the change as either Standard or Emergency.
        i. Standard changes proceed via the regular process.
        ii. Emergency changes follow the Emergency Change procedure.

Categorization

1. **Role:** Change Requestor
2. **Description:**
    a. Examine how the change impacts systems, services, or resources (e.g., Application, Access, Network, or Other).

b. Assign the category (e.g., “Application,” “Network,” etc.) and note any potential risks or resource needs.

Initial Priority Allocation

1. **Role:** Change Requestor
2. **Description:**

   a. Based on the scope and impact, assign a priority level to the change (e.g., 1 = critical, 5 = negligible).

   b. This dictates the order in which changes are reviewed and scheduled.

Registering a Change

1. **Role:** Change Manager
2. **Description:**

   a. Every change is registered using a Change Request (CR) ticket (e.g., in JIRA).

   b. Record basic details: description, objective, benefits, system(s) affected, and change type.

Emergency Change

1. **Role:** Change Manager
2. **Description:**

   a. If the CR is the highest priority and must be done immediately, it becomes an Emergency Change Request.

   b. Swift approval is required. Move to Step 5a for emergency approvals; otherwise, proceed to Step 6 for standard changes.

Emergency Change Approval

1. **Role:** Change Manager
2. **Description:**

   a. Secure verbal or email approval from authorized Change Approvers.

   b. Document the change in a CR post-implementation if time constraints do not allow prior registration.

Process for Implementing a Standard Change

1. **Roles:** Change Requestor → Implementation Team
2. **Description:**

   a. The Implementation Team reviews the CR.

   b. No additional approvals are required for pre-authorized Standard Changes.

   c. Proceed to Step **Assessing, Approving, and Scheduling a Standard Change** for risk assessment and scheduling.

Assessing, Approving, and Scheduling a Standard Change

1. **Role:** Change Manager / Peer Developer / QA
2. **Description:**

   a. Conduct risk assessment and impact analysis to determine potential consequences if implemented or not.

   b. Based on internal rules, alternative or additional approval (e.g., an email from a reviewer or authorized approver) may be required.

   c. Once approved, schedule the change and proceed to build/test.

Process for Implementing a Normal Change

1. **Roles:** Change Requester, Implementation Team
2. **Description:**
   a. The request is sent to the Implementation Team.

   b. Approvals are required for Normal Changes (unlike a fully pre-authorized Standard Change).

Normal Change Approval

1. **Role:** Change Approver
2. **Description:**
   a. Obtain email or documented approval from authorized change approvers.

   b. Changes are implemented only after approvals.

   c. If new CRs are needed, ensure they are raised parallel or upon completion.

Coordinate Build and Test

1. **Role:** Implementation Team
2. **Description:**
   a. Once the CR is approved, build/configure the change in the test environment.

   b. All Standard/Normal changes require testing before deployment.

   c. Keep records of test plans and outcomes.

Build and Test Success (Application Services)

1. **Role:** Implementation Team → QA Team
2. **Description:**
   a. Define test parameters and expected results before development.

   b. QA verifies whether the change meets the success criteria.

   c. If successful, proceed to Step 11 for live implementation.

   d. If unsuccessful, proceed to Step 10 or 13 for rollback or rework.

Inform the Change Initiator

1. **Role:** Implementation Team
2. **Description:**

a. The initiator is notified if the change cannot be implemented for technical or approval reasons.

b. If the deployment fails in test or partial deployment, the CR may be canceled, or a rollback plan may be triggered.

Coordinate Change Implementation (Production)

1. **Role:** Implementation Team
2. **Description:**

    a. After successful build and test, move the change to the live (production) environment.

    b. Implementation Team executes the deployment following approved steps and schedules.

Change Successful

1. **Role:** Implementation Team / QA Team
2. **Description:**

    a. Validate post-deployment that the change is working as intended.

    b. If issues arise, proceed to **Coordinate Rollback**; otherwise, continue to **Post-Implementation Review and Closure**.

Coordinate Rollback

1. **Role:** Implementation Team
2. **Description:**

    a. If the production deployment fails, execute the rollback plan.

    b. Document reasons for failure and any impact observed.

Post-Implementation Review and Closure

1. **Role:** CTO/Appointed Reviewer CTO / Appointed Reviewer CTO/Appointed Reviewer
2. **Description:**

    a. Conduct a post-implementation review to verify the change meets objectives and does not introduce new risks.

    b. If possible, a different individual from the original reviewer (e.g., the QA or a separate manager) should perform this review.

    c. Once completed, close the CR with relevant notes and lessons learned.

Annexure - A

A1 Categorization Guideline

Standard Change

1. Well-known, frequent, and typically low-risk changes (e.g., patching non-critical systems, routine maintenance).
2. Requires the full range of assessments but often has pre-approved status if documented procedures are established.
3. It may still carry high risk based on impact and probability of failure.

Emergency Change

1. This must be done immediately due to a high-priority incident (e.g., a security breach or system crash).
2. Scheduling can be omitted or fast-tracked; customary approvals may be condensed to expedite resolution.

A2 Change Priority Matrix

Change Priority	System / Service Availability Requirement
1	• Changes that impact significant parts of the IT operations, • Changes that impact the services of multiple customers and require pre-defined downtime. • Changes required for resolving security incidents.
2	• Changes that will impact partial services for multiple customers.
3	• Changes will impact parts of the IT operation and all the services of single customers.
4	• Changes that will impact partial services for a single customer • All the Standard Changes.
5	• There is no impact on the System or Service.

Version Details

Version	Version Date	Description of changes	Created By	Approved By	Published By
Version 1.0	Aug 29 2025	Initial Release	Borhan	Linh	Borhan