

Course: Software Requirements Engineering

Week 12: Requirements Change Management

Lecturer: Yimer Amedie (MSc.)

Addis Ababa Science and Technology University

May, 2026

Contents



- Introduction
- Requirements change management (RCM)
- Requirement changes and change control process
- Impact analysis, traceability matrix, and versioning
- Stakeholder roles, communication and documentation
- Challenges and best practices

Figure 1. *The software requirements*

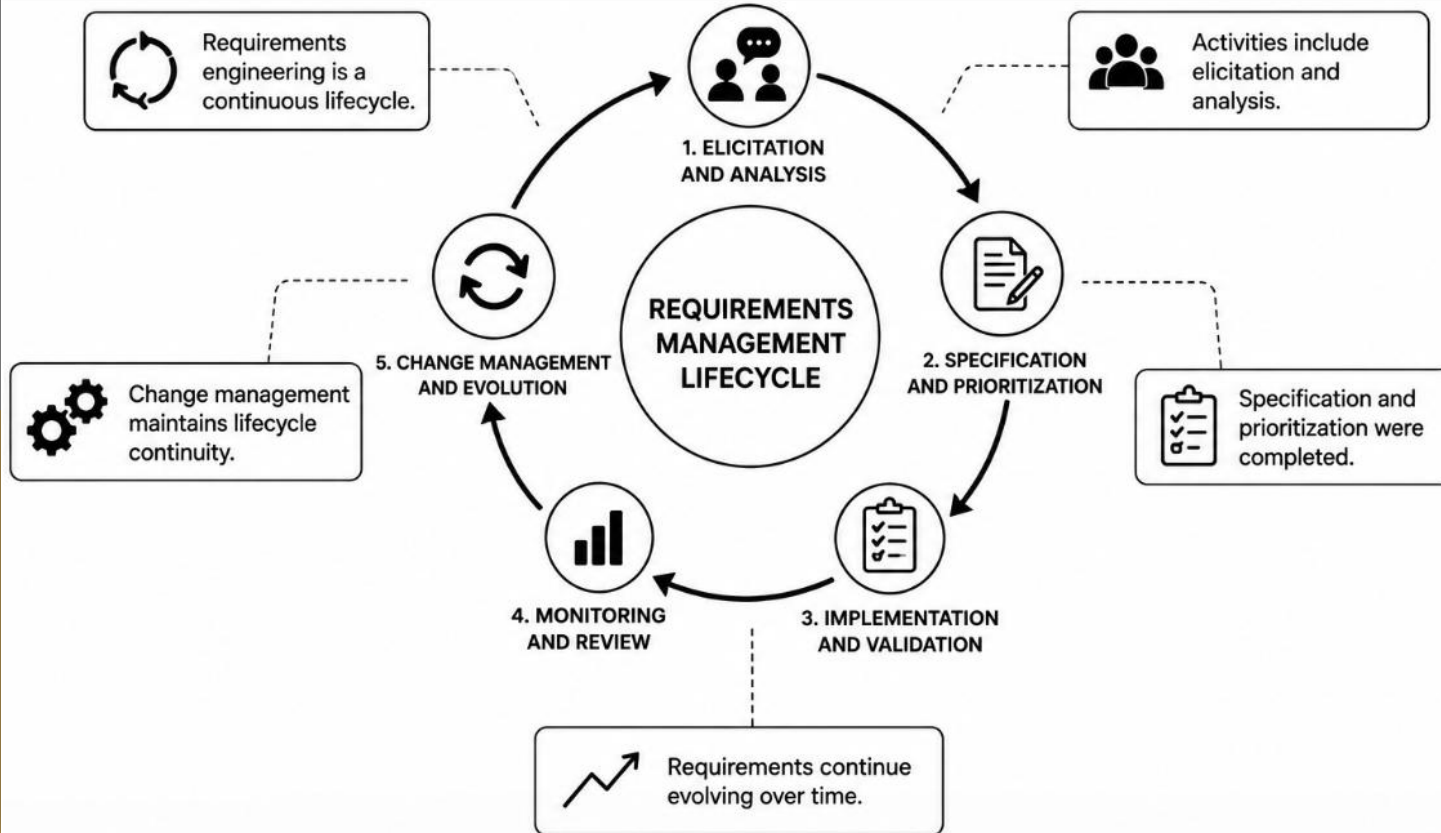
Note. Image generated using Sora by OpenAI (2026).

Learning Outcomes

After completing this lesson, you will be able to:

- Explain the concept of requirements change management
- Identify causes, types, and volatility of requirement changes
- Apply change control, impact analysis, and traceability techniques
- Analyze the effects of requirement changes on cost, schedule, scope
- Use requirement versioning and best practices to support effective change management in software projects.

Introduction



Requirements change management



1

Requirements evolve over time

Business needs, user expectations and technologies change over time, causing requirements to evolve.



2

Change is unavoidable in projects

New insights, market shifts, stakeholder feedback and external factors make change inevitable.



3

Changes affect system development

Requirement changes impact scope, design, schedule, cost, testing and overall product quality.



4

Controlled management is essential

A structured change management process ensures all changes are evaluated, approved, implemented and communicated properly.



5

Supports project success and stability

Effective change management delivers the right product, on time and within budget, leading to satisfied stakeholders and long-term success.



Well-managed changes lead to better decisions, reduced risks, higher quality solutions and successful projects.



Right Requirements.
Right Product. Right Time.

What is Requirement Change?



Modification to existing requirement



Addition of new requirement



Removal of obsolete requirement



Change in priority or scope



Can occur anytime during project



Requirement change is a natural part of projects.

Managing it well ensures better outcomes and successful delivery.

Why Requirement Change?



1 Changing business needs

Business goals, market conditions, and customer needs evolve over time, requiring changes to requirements.



2 Stakeholder feedback and requests

Stakeholders provide new insights, suggestions, or change requests based on their experience and needs.



3 Technological advancements

New technologies and tools create opportunities to improve functionality, performance, and user experience.



4 Legal and regulatory updates

Changes in laws, regulations, and compliance standards require updates to meet legal obligations.



5 Discovery of missing requirements

As analysis and development progress, new understanding may reveal gaps or missing requirements.

Requirement volatility in dynamic environments



Change is natural in any project. Understanding the reasons helps in managing changes effectively and **delivering the right solution.**

Types of Requirement Change



Functional requirement changes



Non-functional requirement changes



Interface requirement changes



Business rule changes



Regulatory and compliance changes

Importance of Change Management



Maintains project control

Ensures that all changes are evaluated and approved, keeping the project aligned with its objectives.



Reduces project risks

Identifies the impact of changes early and helps in mitigating potential risks.



Prevents uncontrolled modifications

Stops unauthorized or unnecessary changes that can derail the project.



Improves communication among stakeholders

Provides a clear process for discussing, reviewing and deciding on changes.



Supports software quality and consistency

Ensures that changes are implemented properly, maintaining the quality and consistency of the software.

Challenges of Requirement Change



Increased project complexity



Budget and schedule overruns



Communication difficulties



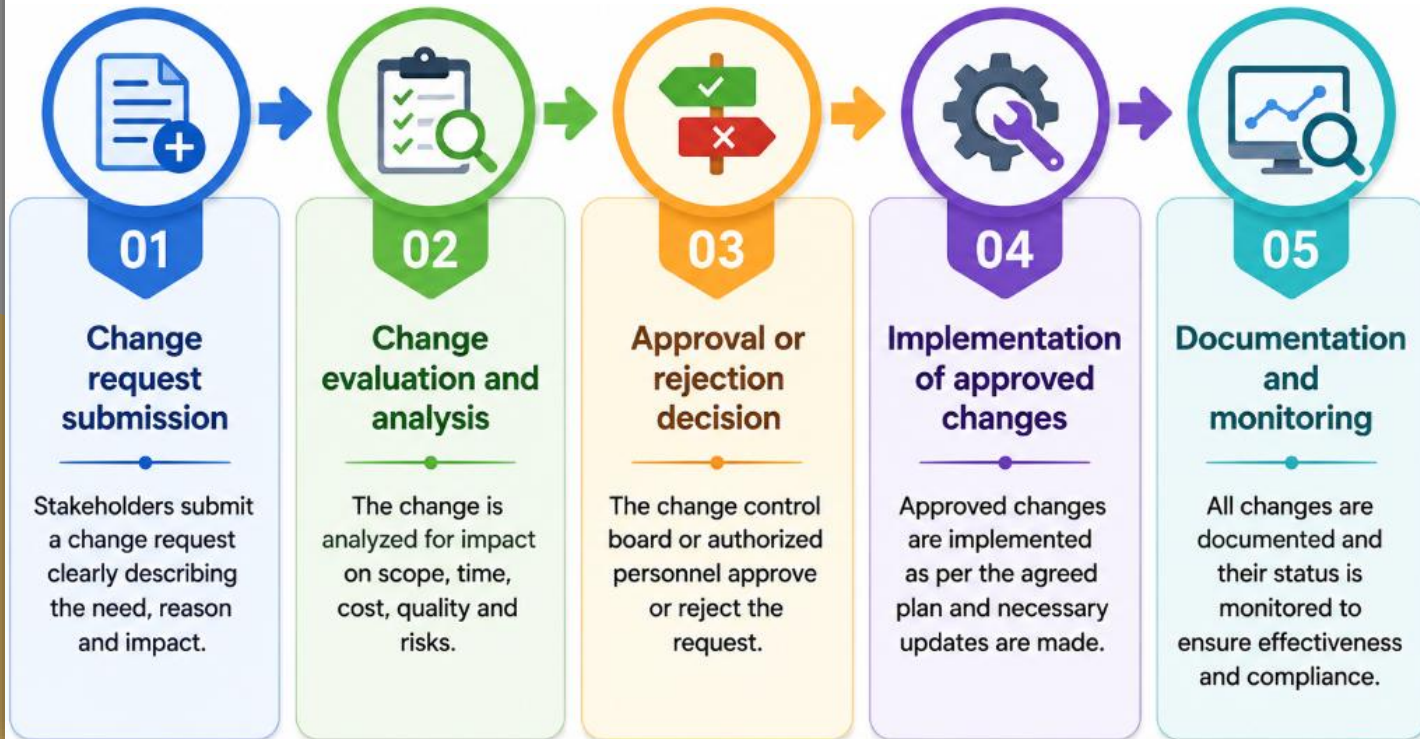
Resistance from stakeholders



Documentation inconsistency risks

Requirement Change Management Process

Change control: A process for reviewing, approving, & managing requirement changes.



Systematic management of requirement modifications through evaluation, approval, and implementation procedures.

Change Request



Formal proposal for modification



Submitted by stakeholders or team



Includes reason for change



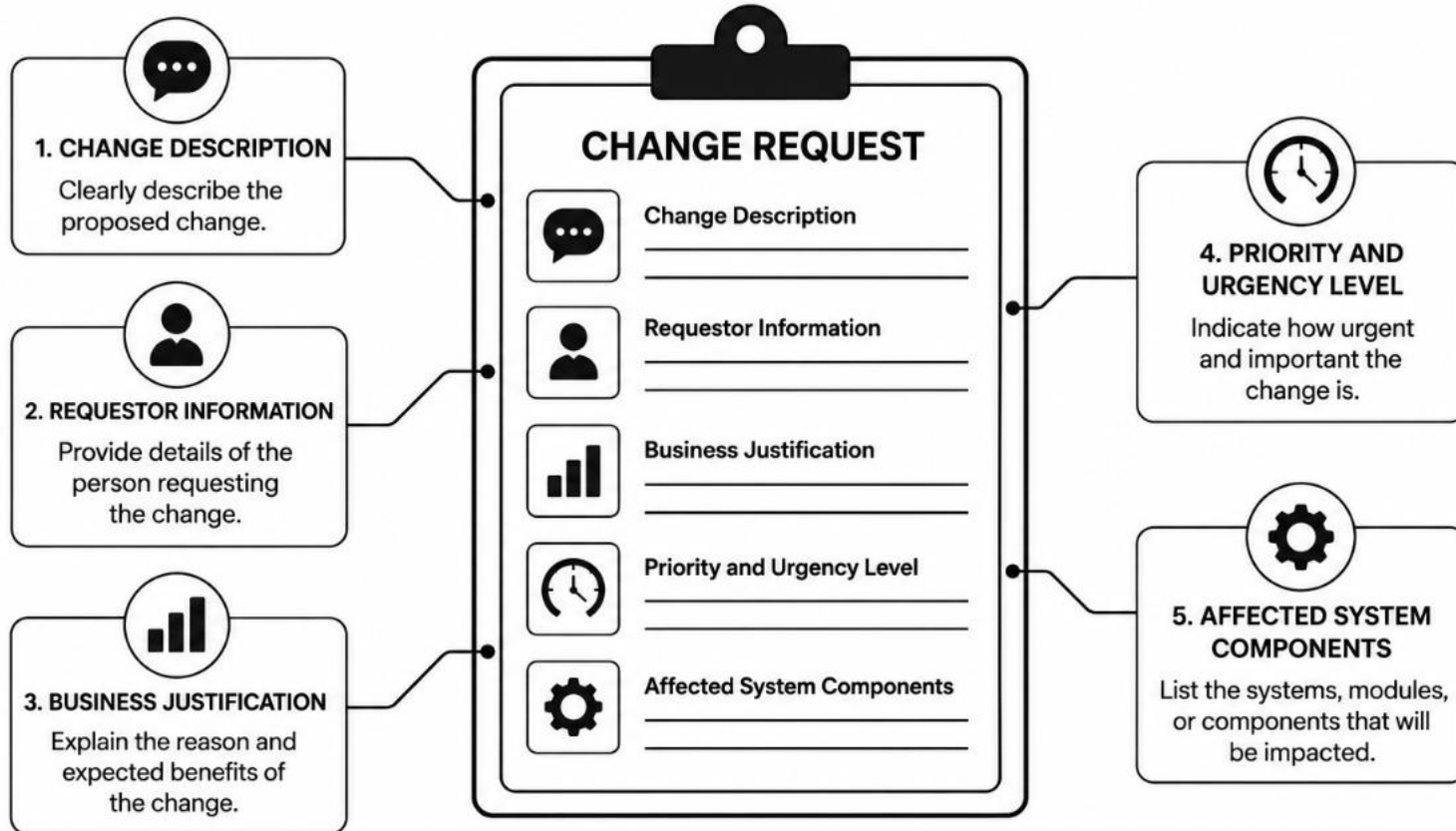
Describes expected benefits



Starts change management process



Change Request Components



Change Control Board (CCB)



Figure 2. *What is CCB?*
Note. Image generated using ChatGPT by OpenAI (2026).

Impact Analysis



Evaluates consequences of changes



Assesses technical impact



Estimates cost and schedule effects



Identifies affected stakeholders



Supports informed decision-making

Cost Impact of Requirement Changes



Additional development effort



Increased testing activities



More resource requirements



Possible rework of completed tasks



Budget adjustment may be needed



Schedule Impact of Requirement Changes



Delays development activities



Affects project milestones



Requires updated timelines



May interrupt ongoing tasks



Impacts delivery deadlines



Scope Creep

When Change is Uncontrolled, Problems Multiply


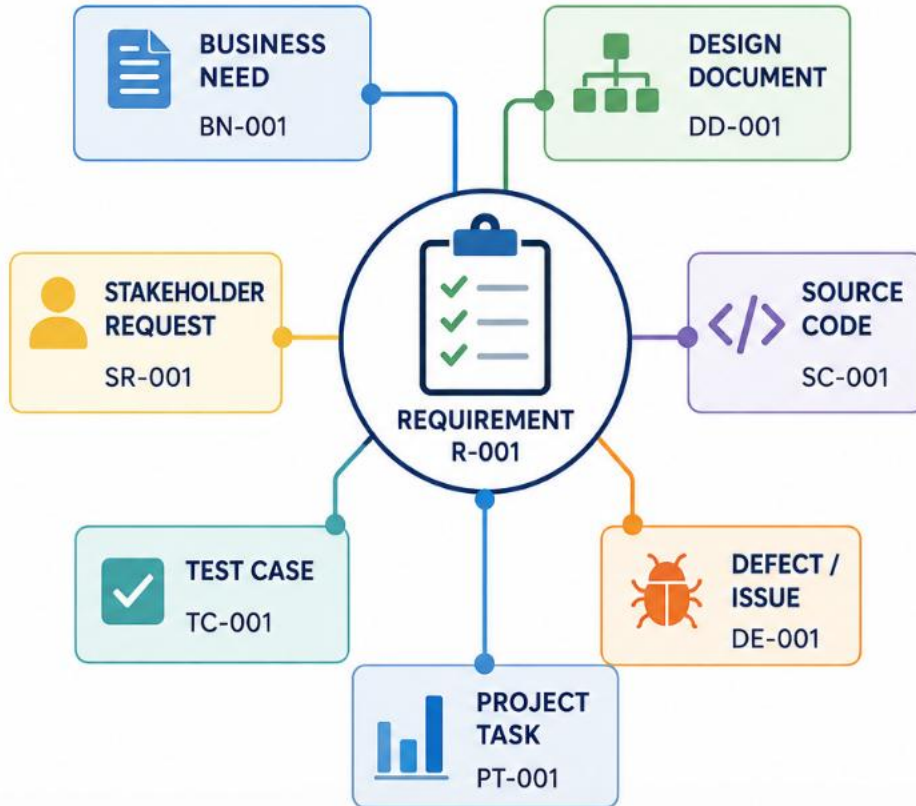
-  Uncontrolled expansion of project scope
-  Caused by unmanaged changes
-  Increases project risks
-  Affects budget and schedule
-  Requires strict change control



Figure 3. Concept of Scope Creep

Note. Image generated using ChatGPT by OpenAI (2026).

Requirement Traceability



Links requirements to project artifacts



Tracks requirement relationships



Supports impact analysis



Helps manage changes effectively



Improves accountability and consistency

Traceability Matrix



Table linking related artifacts



Connects requirements and tests



Supports verification activities



Identifies affected components quickly



Useful for audits and maintenance

TRACEABILITY MATRIX

REQUIREMENTS	DESIGN			TESTS			COMPONENTS
	SRS Section	Design Document	Module	Test Case ID	Test Procedure	Test Result	Component / Module
REQ-001 User Login	3.1	DD-01	MOD-01	TC-01 TC-02	TP-01	✓	Auth Module
REQ-002 Reset Password	3.2	DD-01	MOD-01	TC-03	TP-02	✓	Auth Module
REQ-003 View Dashboard	3.3	DD-02	MOD-02	TC-04 TC-05	TP-03	✓	Dashboard Module
REQ-004 Generate Report	3.4	DD-03	MOD-03	TC-06 TC-07	TP-04	✓	Report Module
REQ-005 User Logout	3.5	DD-01	MOD-01	TC-08	TP-05	✓	Auth Module

✓ Trace Link Exists

— No Direct Link

● Partial Link

Version Control in Requirement Management



Tracks requirement document changes



Maintains revision history



Supports collaboration among teams



Prevents loss of previous versions



Improves change transparency



Revision History

v2.3 May 20, 2024 10:30 AM
Updated by Priya

v2.2 May 15, 2024 04:45 PM
Updated by Rahul

v2.1 May 10, 2024 11:20 AM
Updated by Anika

v2.0 May 05, 2024 09:15 AM
Updated by John

v1.0 Apr 30, 2024 02:00 PM
Created by Sarah

Role of Stakeholders in Change Management



Figure 4. *Role of Stakeholders in Change Management*

Note. Image generated using ChatGPT by OpenAI (2026).

Communication in Requirement Changes



Keeps stakeholders informed



Reduces misunderstandings and conflicts



Supports collaborative decision-making



Ensures transparency in changes



Improves project coordination



Figure 5. *Communication in Change Management*

Note. Image generated using ChatGPT by OpenAI (2026).

Documentation of Requirement Changes



Records all approved modifications



Updates requirement specifications



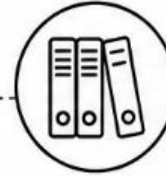
Maintains project consistency

REQUIREMENT CHANGE LOG			
ID	DATE	DESCRIPTION	STATUS
01	05/01/24	Change in login process	Approved
02	05/10/24	Update to user profile requirements	Approved
03	05/15/24	Add password reset feature	Approved
...

APPROVED



Supports future maintenance activities



Provides historical reference information



CHANGE REQUEST



REVIEW



APPROVAL



DOCUMENTATION



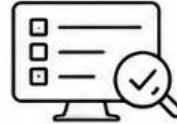
HISTORICAL RECORD

Requirement Change Management Tools



Jira

for issue tracking



IBM DOORS

for requirement management



Azure DevOps

collaboration support



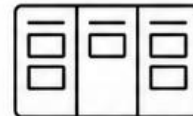
Git

for version control



Trello

for task coordination



Change Management in Agile Development



Figure 6. *Change Management in Agile Development*

Note. Image generated using ChatGPT by OpenAI (2026).

Change Management in Traditional Models



Formal change procedures followed



Extensive documentation required



Sequential development stages affected

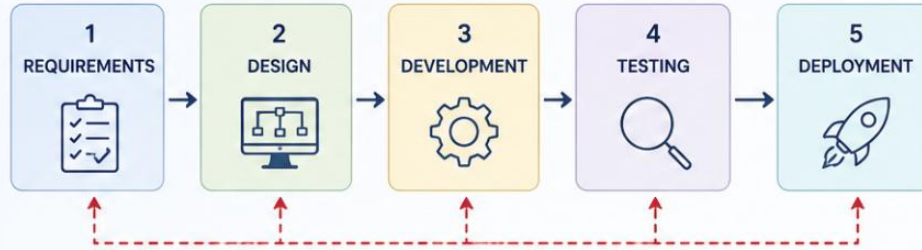


Changes harder in later phases



Strong emphasis on approval control

TRADITIONAL (WATERFALL) DEVELOPMENT



CHANGES ARE COSTLY AND DIFFICULT IN LATER PHASES

CHANGE MANAGEMENT PROCESS



CHANGE APPROVAL

- Justification
- Impact Analysis
- Cost
- Schedule
- Resources

APPROVED

Risk of Change Management



Project
delays and
overruns



Increased
software
defects



Stakeholder
dissatisfaction



Loss of
project
control



Reduced
software
quality

Best Practices

Establish formal change procedures

Define a structured process for submitting, reviewing, and approving changes

Maintain clear documentation

Keep records of all changes, rationale, and version history accessible

Perform impact analysis consistently

Assess scope, cost, and schedule implications before any change is approved

Involve stakeholders regularly

Engage relevant parties early to align expectations and reduce rework

Use traceability and management tools

Leverage software to track links between requirements, changes, and deliverables

Example – OLMS

Initial requirement

Online registration — the baseline scope defined at project start

New request added

Mobile learning support requested by stakeholders mid-project

Security requirements added

Additional security constraints introduced to address compliance needs

Impact analysis performed

Cost and schedule effects evaluated before proceeding with changes

Changes approved and implemented

All changes signed off and successfully integrated into the system

Future Trends in Requirement Change Management



Increased automation
in change tracking



AI-supported
impact analysis tools



Cloud-based
collaboration platforms



Real-time stakeholder
communication



Greater integration
with DevOps practices



Figure 7. *Future trends in requirement change management*

Note. Image generated using ChatGPT by OpenAI (2026).

Summary

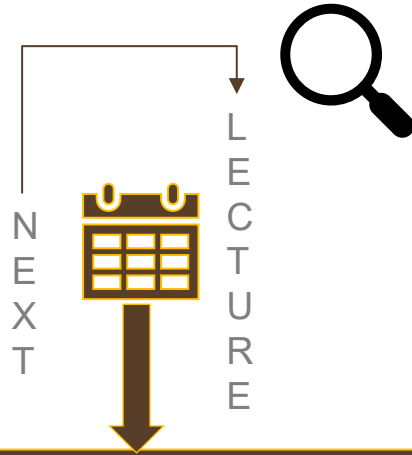
In software
projects

- ✓ Requirement changes are unavoidable
- ✓ Structured management is essential
- ✓ Impact analysis supports decisions
- ✓ Traceability improves control and consistency
- ✓ Effective practices increase project success

References

1. Beatty, K. W. (2013). Software Requirements (3rd ed.). Washington: Microsoft Press.

Thank You!



**Tools & Agile Requirements
Engineering**