

CLM Implementation Checklist

Implementing contract lifecycle management (CLM) involves several key steps to ensure successful deployment and adoption. Once a software vendor has been selected, **here are the general steps typically involved in a CLM implementation plan:**

 Plan	1. Project Planning • Resource Allocation: Identify and assign roles and responsibilities to team members from both the customer and vendor sides. • Project Pre-work: Exchange preliminary information on high-level project objectives, requirements, milestones and procedures.
	2. Current State Assessment • Contract Inventory: Compile an inventory of existing contracts and agreements across the organization. • Process Mapping: Analyze current contract management processes and related processes to identify pain points, inefficiencies, and areas for improvement.
 Design	1. Project Initiation • Kickoff Meeting: Gather stakeholders from both the customer organization and the vendor to align on goals, scope, timelines, and resources. • Project Plan: Develop a detailed project plan outlining tasks, responsibilities, milestones, and timelines.
	2. Requirements Gathering • Stakeholder Interviews: Conduct interviews with key stakeholders to understand their requirements and expectations from the CLM system. • Documentation Review: Review existing contract templates, workflows, and policies to inform system configuration.
 Build	1. System Configuration And Customization • Iterative Configuration: Work with the vendor to configure the CLM system based on gathered requirements and validate those configurations. • Customization (if needed): Develop any necessary customizations or integrations to align the CLM system with existing IT infrastructure and business processes.
	2. Data Migration • Data Mapping: Map data fields from existing systems to the new CLM system. • Data Cleansing: Clean and validate data to ensure accuracy and consistency. • Migration Execution: Transfer contract data and documents from legacy systems to the new CLM system.



Test and Train

1. Testing

- **System Testing:** Customer and vendor conduct comprehensive testing to validate system functionality, integrations, and data accuracy.
- **User Acceptance Testing (UAT):** Engage end-users to test the system against real-world scenarios to ensure it meets their needs.

2. Training

- **System Testing:** Customer and vendor conduct comprehensive testing to validate system functionality, integrations, and data accuracy.
- **User Acceptance Testing (UAT):** Engage end-users to test the system against real-world scenarios to ensure it meets their needs.



Deploy

1. Deployment

- **Deployment Plan:** Prepare a deployment plan outlining the steps and timeline for rolling out the CLM system.
- **Go-Live and Hypercare Support:** Provide support during the initial launch phase to address any immediate issues or questions from users.

2. Ongoing Support And Maintenance

- **Support Plan:** Establish a support framework to address ongoing maintenance, updates, and user support needs.
- **Continuous Improvement:** Continuously monitor system performance and user feedback to drive ongoing improvements and updates.

3. Post-Implementation Review And Optimization

- **Feedback Collection:** Gather feedback from users and stakeholders post-implementation.
- **Performance Evaluation:** Evaluate systems performance against predefined metrics and KPIs.
- **Optimization:** Identify areas for improvement and implement enhancements to maximize the system's effectiveness.



Each of these steps is crucial for a successful CLM implementation, ensuring that the system meets organizational requirements, enhances efficiency, and facilitates better contract management practices.