AI Coding Governance Framework

This governance framework establishes the standards, controls, and oversight required for building, deploying, and maintaining applications developed using AI-generated code, AI development copilots, and AI-driven logic. It ensures all AI-assisted development aligns with industry standards for security, privacy, accessibility, accuracy, transparency, and operational integrity.

Some of the benefits of using AI in Coding include the dramatic reduction of time to develop working solutions, ability of non-coders or those with limited IT experience to create functional applications and shift focus from syntax errors to creative problem solving.

The goal is to ensure that all solutions are secure, compliant, accessible, consistent, and aligned with organizational policies.

Definition

- **Artificial Intelligence** refers to computational systems capable of performing tasks that typically require human intelligence.
- **Generative AI** refers to AI models capable of producing original outputs such as text, code, images, summaries, workflows, or recommendations.
- **AI Code Generation** is the process by which an AI system produces software code, scripts, queries, API calls, UI components, tests, or configuration files based on human instructions or prompts.
- **AI-Assisted Software Development** is an AI-driven development approach where software is created by prompting Large Language Models (LLMs) in natural language rather than writing code manually. AI-Assisted Software Development is considered a significant shift in software development that leverages the increasing capabilities of LLMs to make coding more accessible and fast-paced.
- **AI prompt** is any instruction, question, or input provided to an AI system to generate an output.
- **Human-in-the-Loop** ensures humans maintain full control and review over any AI-generated artifacts.
- **Explainability** means the AI-produced code, reasoning, or output must be understandable to humans.
- **Citizen Developer** is a non-IT employee who creates applications, automations, or digital solutions using low-code, no-code or AI platforms

Objectives

- Enable citizen developers and business staff to build applications using AI generated code safely and effectively.
- Ensure all AI generated coded solutions follow the city's security, privacy, accessibility, and data protection standards and comply with IT Resource and AI Policy
- Ensure AI-generated code is secure, reviewed, tested, and compliant before production deployment.
- Maintain accountability, explainability, and quality in applications built using AI assistance.
- Reduce operational, security, and legal risks associated with AI-generated output.
- Establish lifecycle governance for AI-assisted development from requirements → code generation → testing → deployment → maintenance.
- Enable responsible AI adoption while maintaining trust, safety, and transparency.

Scope

This governance applies to:

- All applications where AI generates any portion of code, scripts, logic, SQL, APIs, or configuration.
- All staff using AI tools such as code generation, generative AI IDEs, AI workflow generators, or AI analysis engines.
- All deployments into development, test, staging, and production environments.

Development Requirements

- Engage IT early during the planning phase to ensure alignment on architecture, security, implementation and support strategy.
- Human developers retain full ownership and accountability for all AI-generated code. No AI-assisted component may be promoted to production without appropriate human peer review, testing, and security validation in coordination with IT.
- IT will provide guidance on approved technologies and enterprise standards. The project code, build process, code repository, development tooling (AI or traditional), and all associated systems must be reviewed and approved by IT to ensure alignment with enterprise security, architecture, and compliance standards.

- Developers should document the AI tools used, location of data, and format of data used.
- Developers are responsible for producing complete functional and technical documentation for all the generated code. Use of generative AI to write documentation is permissible, provided the content is reviewed and validated by humans.
- All AI-generated applications, user interfaces, documentation, and content must comply with City UI design standards and brand guidelines, meet ADA/WCAG 2.1 AA accessibility requirements, and adhere to the City's Information Security Policy.

Integration & Data Standards

- Only IT-approved application connectors, APIs, and integration mechanisms may be used for system integration and data exchange
- Sensitive or restricted data, as defined in the City's Enterprise Data Strategy and Governance Plan, cannot be used.
- All integrations must log activity and errors.
- Enterprise data should have a designated system of record. Replication, caching, or synchronization of data from that system is permitted for performance, analytics, or integration purposes, provided the original system remains the authoritative source and data governance standards are followed.

Change Management & Release Requirements

- Only authorized IT administrators will deploy code to production or implement configuration changes in production environment.
- Release notes and documentation must accompany code updates.
- All changes must follow established IT change management approval and deployment procedures.

Validation & Security Requirements

- Standardize AI coding platform - essential to ensure it meets baseline security, privacy, and reliability requirements before it is trusted in the development process.
- Centralize Code Repository & Version Control – use a centralized code repository and mandate version control. Using a standard code repository with version control is essential for maintaining visibility, accountability, and control over software changes. It ensures that all code—including AI-generated code—is tracked, reviewed, and auditable, enabling teams to understand what changed, when, and

why. All mergers into a production branch must be submitted via pull request for consideration and approval by IT.

- Code Review – remains a critical control in AI-assisted development because it ensures that automatically generated code meets security, quality, and compliance standards before deployment. AI generated code can introduce insecure code and vulnerabilities into applications. All codes will be reviewed by IT prior to launch. In some cases, revisions will be necessary. IT will coordinate these with the original developer.
- Design Review – All applications and content provided to the public must conform to the City's brand guidelines and general design best-practices and should contribute to a cohesive experience for the user in engaging with City services. The design of all applications and content will be reviewed by IT prior to launch. Any necessary changes will be coordinated with the original developer.
- Secrets Management - ensures that sensitive credentials, API keys, and tokens are stored, rotated, and accessed securely, preventing accidental exposure in code or AI-assisted workflows. Coding agents should not have direct access to sensitive environment variables by default. Where necessary, access must be mediated through approved secrets management solutions, limited by least-privilege principles, and restricted to specific use cases.
- Software Bill of Materials – The original developer should provide a complete inventory of all components, libraries, and dependencies used in a project, including AI-generated or third-party code. This visibility enables proactive vulnerability management, supports compliance, and reduces the risk of supply-chain attacks.
- Access Reviews – IT will conduct regular access reviews to ensure that only authorized individuals can view, modify, or deploy code and related resources. This control reduces the risk of accidental or malicious changes, enforces the principle of least privilege, and provides oversight for AI-assisted development.
- Web Application Scans – IT will verify that the deployed site is free from common client-side vulnerabilities and configuration gaps. Scans help identify issues such as unsafe scripts, exposed files, missing security headers, and weak content security policies that can arise during rapid or AI-assisted development.
- Load Testing – confirms that a website or application can handle expected peak traffic by simulating real user requests. IT will perform a load test on all projects prior to launch. In cases where performance improvements are needed, IT will coordinate these with the original developer.

- Cross Browser Testing – verifies that the public website or application loads and functions according to expectations across all supported browsers, operating systems, and screen sizes. This is a necessary step to uncover bugs and design issues which only appear under specific circumstances. IT will perform cross browser testing on all websites and web applications prior to launch. In cases where fixes are needed, IT will coordinate these with the original developer.
- Accessibility Auditing – All media distributed to the public, including websites and applications, must be compliant with WCAG 2.1 AA guidelines. The original developer will be expected to make a best effort to achieve compliance during the development process. IT will provide tooling for easily assessing compliance with many of the requirements. Since automated testing cannot identify all accessibility issues, after initial development is complete, an accessibility audit will also be performed, either by IT or an outside vendor. In cases where fixes are needed, IT will coordinate these with the original developer.
- Reassessment – Updates to the project code may require reassessment of any or all of the above validation items. It is recommended to manage any content separately from code so that content edits do not result in reassessment.

Monitoring, Maintenance & Continuous Improvement

- AI coding platform should have audit trails and if possible forward system and audit logs to a Security Information Event Management (SIEM) platform to detect anomalies such as failed authentication attempts.
  Monitor AI models, data inputs, and outputs to detect unusual patterns, or unexpected behavior.

Roles & Responsibilities

- Citizen Developers / Business Developers
  - Complete required training before building.
  - Use IT-approved AI coding tools, templates, components and technologies.
  - Document AI application logic, and data flows.
  - Validate all AI-generated code.
  - Ensure code integrity, readability, and maintainability.
  - Report anomalies or unexpected AI behavior.
- IT Department
  - Perform risk assessments, review complex logic and security implications.
  - Ensure compliance with security, privacy, and other IT standards.

- Approve deployment to production.
- Manage infrastructure, environments, and integration points.
- Monitor system performance, and logs.

AI/Citizen Developer Project Lifecycle

The following process applies to all citizen developers and business-led development projects to ensure security, accessibility, supportability, and alignment with City technology standards:

1. **Engage IT Early** – IT collaborates with the business during project planning to define scope, architecture, and compliance needs.
2. **Repository Setup** – IT provisions secure code repositories and version control structures.
3. **Build with Approved Technologies** – Development must use IT-approved tools, frameworks, AI platforms, and design standards.
4. **Design & Code Review** – IT conducts formal design and code reviews to ensure quality, security, and alignment with enterprise architecture.
5. **Environment & Pipeline Setup** – IT configures CI/CD pipelines, development environments, testing infrastructure, and production environments.
6. **Revisions & Updates** – Developers and IT collaborate to revise designs and code based on review feedback.
7. **Comprehensive IT Testing**
    a. Cross-browser testing
    b. Performance/load testing
    c. Accessibility audit (WCAG 2.1 AA)
    d. Security scanning and penetration testing
    e. Fixes coordinated with developers and/or vendors
8. **Production Deployment** – Only authorized IT administrators perform production deployments following change management standards.
9. **Launch & Post-Launch Support** – Application is released and IT ensures ongoing monitoring and operational readiness.