CI/CD Checking Requirements and Guidelines

* **Automated Validation Requirements**
  + Implement mandatory build verification before allowing check-ins
  + Configure automated tests unit/integration to run as part of the gate
  + Ensure all validations must pass before code is accepted into the repository
  + Implement rolling builds on the trunk branch to detect issues early
* **Manual Pre-Gate Quality Checks**
  + Verify code formatting standards are met
  + Remove any commented-out code before submission
  + Check for proper encoding and line endings across platforms
  + Validate against organizational coding guidelines
  + Keep changes small and focused
  + Maintain atomic commits that can be easily reverted if needed
* **Testing Requirements**
  + Enforce passing unit tests before acceptance
  + Run automated validation checks on the server
  + Perform local testing before submitting changes
  + Include regression testing for critical components
  + Configure the system to reject failing builds automatically
  + Provide clear feedback when validation fails
  + Document common failure scenarios and resolutions

CI/CD Build Pipeline

* **Source Retrieval**
  + Clone/download source code from version control
  + Validate repository integrity
  + Create isolated workspace for build
* **Dependency Restoration**
  + Install project dependencies
  + Resolve transitive dependencies
  + Cache frequently-used packages
  + Verify dependency integrity
* **Hash Validation**
  + Calculate checksums of critical files
  + Compare against stored hash values
  + Verify dependency integrity
  + Fail build on hash mismatches
* **Unit Testing**
  + Execute individual component tests
  + Report test coverage metrics
  + Capture test execution logs
  + Fail build on test failures
* **Integration Testing**
  + Test component interactions
  + Verify system behavior
  + Execute end-to-end scenarios
  + Validate API contracts
* **Monitoring Setup**
  + Configure application metrics collection
  + Set up performance monitoring
  + Establish health check endpoints
  + Configure alerting thresholds
* **Logging Configuration**
  + Set up logging frameworks
  + Configure log levels
  + Define log retention policies
  + Establish log routing rules
* **Packaging**
  + Create deployment artifacts
  + Generate version manifests
  + Apply digital signatures
  + Prepare for distribution