



Course: IT Project Management
Lecture 8: Project Quality Management

Lecturer: Rambu Yetti Kalaway

Learning Objectives

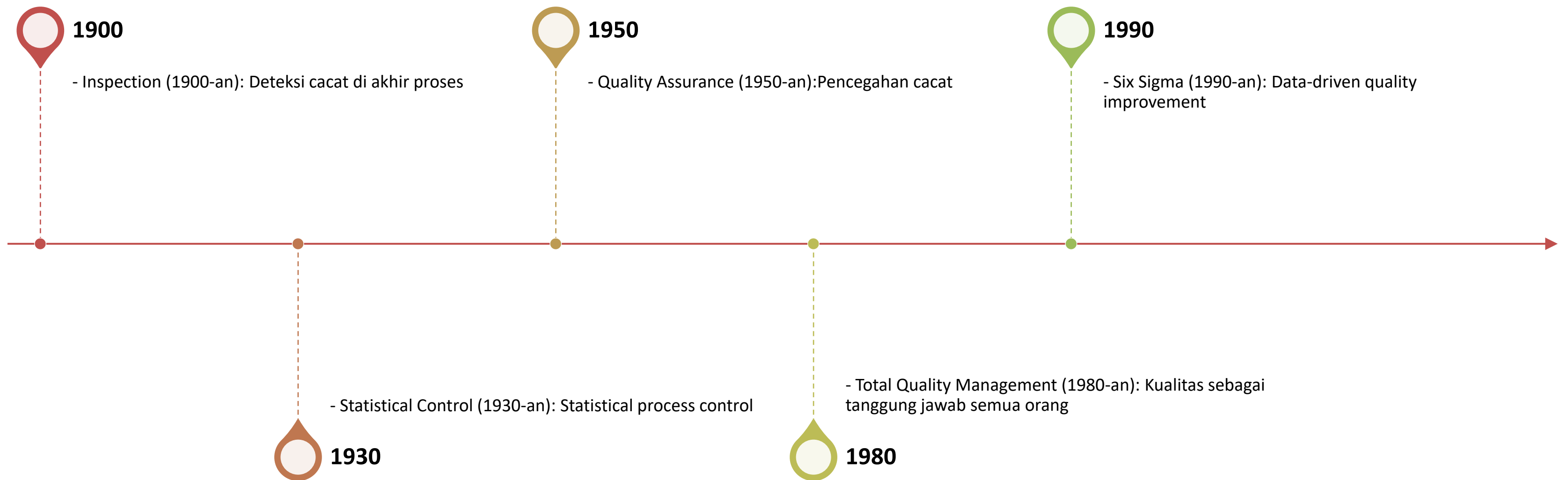
- - Memahami filosofi dan prinsip dasar manajemen kualitas
- - Menguasai proses quality management menurut PMBOK Guide 7 & 6
- - Mampu menerapkan quality planning untuk proyek TI
- - Mengimplementasikan quality assurance dan quality control
- - Menggunakan tools dan teknik quality management modern
- - Menerapkan continuous improvement dalam proyek TI




Definisi Project Quality Management

- - PMBOK Guide 7: "Proses untuk mengidentifikasi requirement kualitas dan/atau standar untuk proyek dan deliverablesnya" (PMI, 2021)
- - ISO 9000: "Tingkat dimana sekumpulan karakteristik inherent memenuhi requirement"
- - Fokus: Conformance to requirements dan fitness for use


Evolusi Filosofi Kualitas





Cost of Quality (COQ)

- - Prevention Costs: Training, process documentation, planning
- - Appraisal Costs: Testing, inspection, audits
- - Internal Failure Costs: Rework, scrap, downtime
- - External Failure Costs: Warranty, liability, lost business (Kerzner, 2022)



Perbedaan Quality vs Grade

- - Quality: Derajat pemenuhan requirement
- - Grade: Kategori atau klasifikasi berdasarkan karakteristik teknis
- - Contoh TI:
 - - High grade, low quality: Fitur lengkap tapi banyak bug
 - - Low grade, high quality: Fitur sederhana tapi bekerja sempurna



Prinsip Kualitas Modern

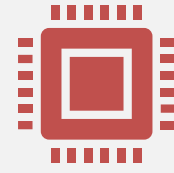
- - Customer Focus: Memahami dan memenuhi kebutuhan user
- - Leadership: Komitmen manajemen terhadap kualitas
- - Engagement of People: Melibatkan seluruh tim
- - Process Approach: Management proses yang terintegrasi
- - Improvement: Continuous improvement berkelanjutan
- - Evidence-based Decision Making: Data-driven decisions
- - Relationship Management: Partner dengan vendor berkualitas



Stakeholder dalam Quality Management

- - Customer/End-User: Penerima akhir yang menentukan fitness for use
- - Project Sponsor: Penyedia dana dengan ekspektasi kualitas
- - Project Team: Pelaksana proses quality assurance
- - Quality Assurance Team: Independent quality verification

Regulatory Environment



- ISO Standards: ISO 9001, ISO/IEC 25010 (Software Quality)



- Industry Standards: IEEE, SEI CMMI

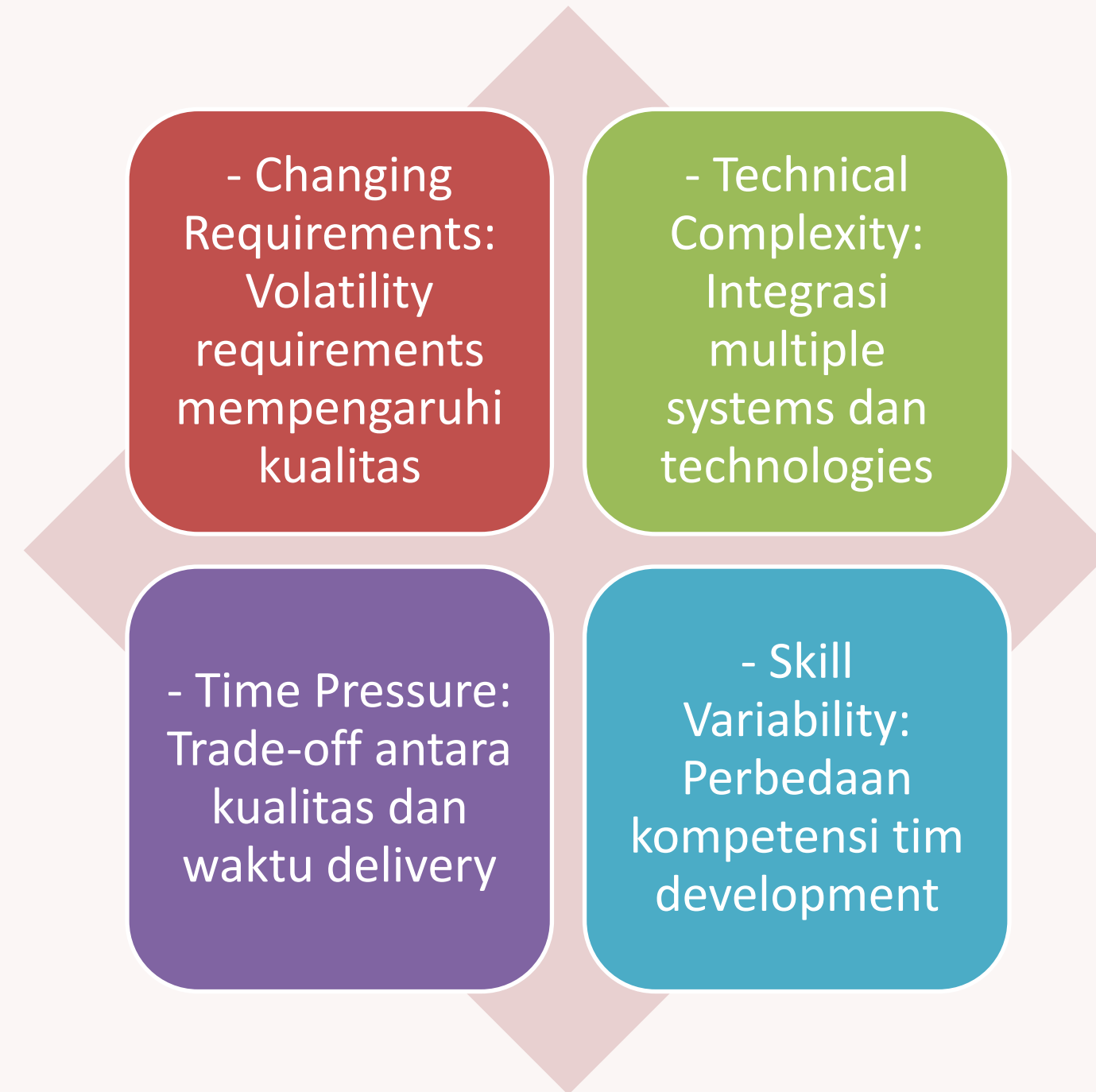


- Legal Requirements: Data privacy, accessibility



- Corporate Standards: Organizational quality policies

Tantangan Quality Management di Proyek TI



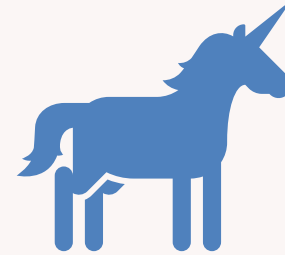
Business Case untuk Quality



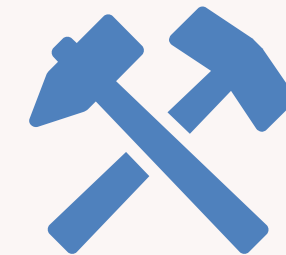
- Increased Customer Satisfaction:
User experience yang baik



- Reduced Costs: Less rework dan
maintenance



- Enhanced Reputation: Brand
image dan competitive advantage



- Higher Productivity: Efficient
processes dan fewer defects

Plan Quality Management - Overview

- - Definisi: Proses mengidentifikasi requirement kualitas dan/atau standar untuk proyek dan deliverablesnya
- - Output: Quality Management Plan, Quality Metrics, Quality Checklists
- - Tujuan: Menyediakan arahan untuk quality assurance dan control





Input Quality Planning

- - Project Charter: Visi dan tujuan kualitas tingkat tinggi
- - Requirements Documentation: Kebutuhan kualitas spesifik
- - Risk Register: Risks terkait kualitas
- - Stakeholder Register: Ekspektasi kualitas stakeholder

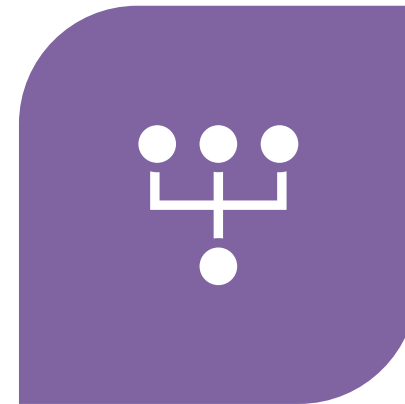
Quality Planning Tools: Cost-Benefit Analysis



- PREVENTION BENEFITS: REDUCED
REWORK, HIGHER PRODUCTIVITY



- APPRAISAL BENEFITS: EARLY
DEFECT DETECTION



- CONFORMANCE COSTS:
TRAINING, PROCESSES,
EQUIPMENT



- NON-CONFORMANCE COSTS:
REWORK, LIABILITIES, LOST
REPUTATION



Quality Planning Tools: Benchmarking

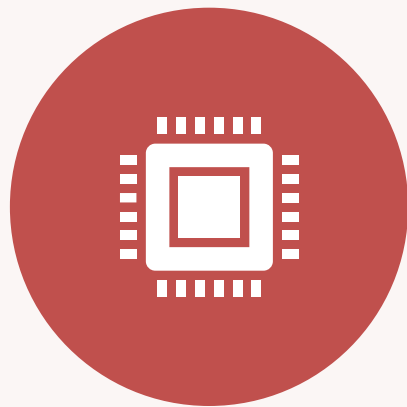
- - Competitive Benchmarking: Membandingkan dengan kompetitor
- - Functional Benchmarking: Membandingkan proses dengan industri lain
- - Internal Benchmarking: Membandingkan dengan proyek internal
- - Best Practices: Adopsi practices terbaik industri

Quality Planning Tools: Design of Experiments

- - Statistical Approach: Mengidentifikasi faktor yang mempengaruhi variabel
- - Process Optimization: Menemukan parameter proses optimal
- - Resource Efficiency: Memaksimalkan output dengan resources minimal
- - Contoh TI: Optimasi database configuration untuk performance



Quality Metrics Development



- PROCESS METRICS: DEFECT DENSITY, TEST COVERAGE, CODE COMPLEXITY



- PRODUCT METRICS: RELIABILITY, PERFORMANCE, USABILITY



- PROJECT METRICS: SCHEDULE ADHERENCE, BUDGET COMPLIANCE

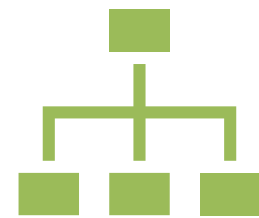


- LEADING VS LAGGING INDICATORS: PREDICTIVE VS HISTORICAL MEASURES

Quality Management Plan Components



- Quality Objectives: Specific, measurable quality goals



- Roles and Responsibilities: Quality-related assignments



- Quality Standards: Applicable standards dan requirements



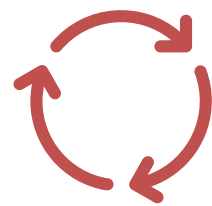
- Quality Control Activities: Inspection dan testing procedures



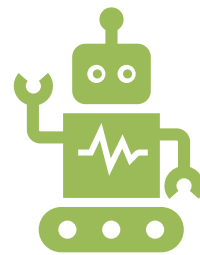
Quality Checklists dan Templates

- - Requirements Checklist: Kelengkapan dan kejelasan requirements
- - Design Checklist: Compliance dengan architectural standards
- - Code Review Checklist: Coding standards dan best practices
- - Testing Checklist: Test coverage dan completion criteria

Modern Quality Approaches



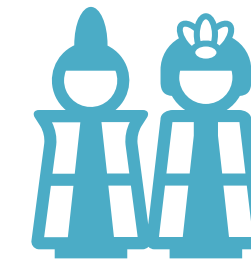
- Shift-Left Testing: Testing early dalam development lifecycle




- Continuous Testing: Automated testing throughout pipeline




- Quality Gates: Checkpoints untuk proceed ke phase berikutnya



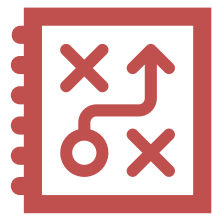
- Definition of Done: Kriteria penyelesaian yang disepakati



Tailoring Quality Planning

- - Agile Projects: Iterative quality planning per sprint
 - - Regulated Environments: Strict compliance requirements
 - - Innovation Projects: Flexible approaches untuk experimentation
 - - Legacy Integration: Additional quality considerations
- 

ISO 9001:2015 Quality Management Systems



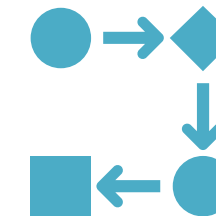
- Process Approach: PDCA (Plan-Do-Check-Act) cycle



- Risk-based Thinking: Proactive risk management




- Context Awareness: Understanding organizational context



- Continuous Improvement: Ongoing enhancement of processes

ISO/IEC 25010 Software Quality Model

- - Functional Suitability: Functional completeness, correctness, appropriateness
 - - Performance Efficiency: Time behavior, resource utilization
 - - Compatibility: Co-existence, interoperability
 - - Usability: Appropriateness, learnability, user error protection
 - - Reliability: Maturity, availability, fault tolerance
 - - Security: Confidentiality, integrity, non-repudiation
 - - Maintainability: Modularity, reusability, analyzability
 - - Portability: Adaptability, installability, replaceability
- 



CMMI (Capability Maturity Model Integration)

- - Maturity Levels: Initial, Managed, Defined, Quantitatively Managed, Optimizing
- - Process Areas: Requirement management, project planning, verification
- - Appraisal Methods: Benchmarking organizational capability
- - Benefits: Process standardization dan improvement

Six Sigma Methodology



- DMAIC: Define, Measure, Analyze, Improve, Control



- DMADV: Define, Measure, Analyze, Design, Verify



- Statistical Tools: Control charts, hypothesis testing, regression

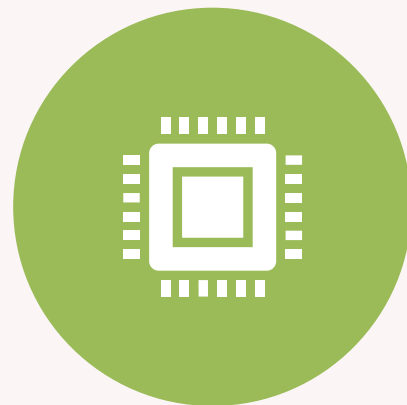


- Quality Goals: 3.4 defects per million opportunities

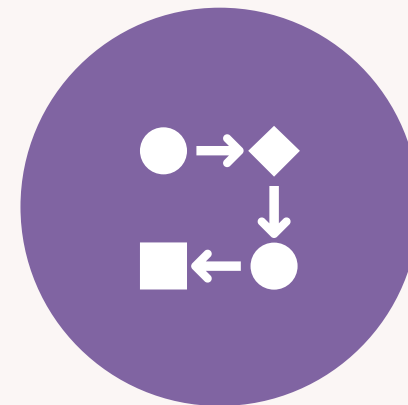
Agile Quality Frameworks



- SCRUM QUALITY PRACTICES:
DEFINITION OF DONE, SPRINT
REVIEWS, RETROSPECTIVES



- XP (EXTREME PROGRAMMING):
TEST-DRIVEN DEVELOPMENT, PAIR
PROGRAMMING



- KANBAN: VISUALIZATION, WIP
LIMITS, CONTINUOUS FLOW



- LEAN SOFTWARE DEVELOPMENT:
ELIMINATE WASTE, AMPLIFY
LEARNING

Testing Standards dan Frameworks



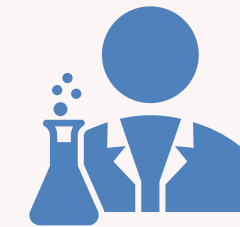
- ISTQB Standards: International Software Testing Qualifications Board



- Test Maturity Model: Assessing testing process maturity



- Automated Testing Frameworks: Selenium, Cypress, Appium



- Performance Testing Standards: Load testing, stress testing protocols

Manage Quality - Overview

- - Definisi: Proses menerapkan aktivitas quality assurance dan mengaudit hasil quality control
- - Fokus: Process improvement dan defect prevention
- - Output: Quality reports, change requests, updates



Quality Assurance vs Quality Control

- - Quality Assurance: Process-oriented, preventive, seluruh tim
- - Quality Control: Product-oriented, detective, tim khusus
- - Hubungan: QA establishes processes, QC validates output
- - Contoh TI:
 - - QA: Code review process, training
 - - QC: Testing, inspection



Quality Assurance Tools: Audits

- - Process Audits: Compliance dengan established processes
- - Product Audits: Conformance dengan requirements
- - System Audits: Overall quality system effectiveness
- - Audit Cycle: Planning, execution, reporting, follow-up

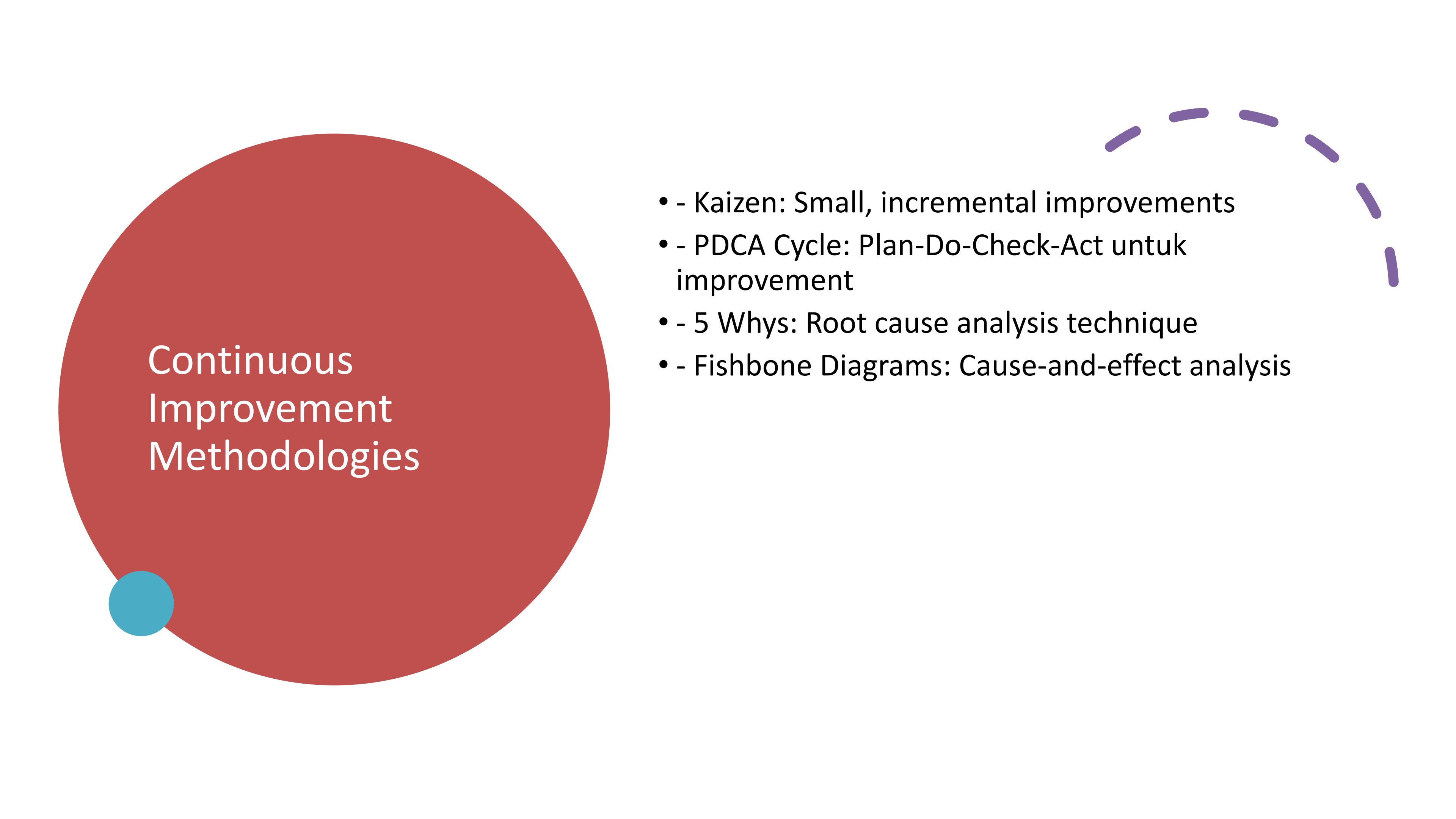
Quality Assurance Tools: Process Analysis

- - Root Cause Analysis: Identifikasi underlying causes of defects
- - Process Mapping: Visualisasi alur proses saat ini
- - Bottleneck Analysis: Identifikasi constraints dalam proses
- - Waste Analysis: Eliminasi aktivitas tidak bernilai tambah



Statistical Tools untuk Quality Assurance

- - Control Charts: Monitoring process stability
- - Pareto Charts: Prioritisasi masalah berdasarkan frequency
- - Histograms: Distribusi data proses
- - Scatter Diagrams: Relationships antara variables



Continuous Improvement Methodologies

- - Kaizen: Small, incremental improvements
- - PDCA Cycle: Plan-Do-Check-Act untuk improvement
- - 5 Whys: Root cause analysis technique
- - Fishbone Diagrams: Cause-and-effect analysis


Quality Assurance dalam Agile

- - Sprint Retrospectives: Regular process improvement
- - Definition of Ready: Kriteria sebelum work dimulai
- - Team Reflection: Continuous improvement practices
- - Process Adaptation: Adjusting processes berdasarkan feedback



A large red circle on the left side of the slide, partially cut off by the edge.

Quality Metrics untuk Assurance

- - Process Performance Metrics: Cycle time, lead time, throughput
 - - Quality Trends: Defect trends over time
 - - Customer Satisfaction: Net Promoter Score, user feedback
 - - Team Performance: Velocity, capacity utilization
- 
- A decorative purple dashed line in the bottom right corner, consisting of several short segments that curve upwards and to the right.

Quality Improvement Initiatives



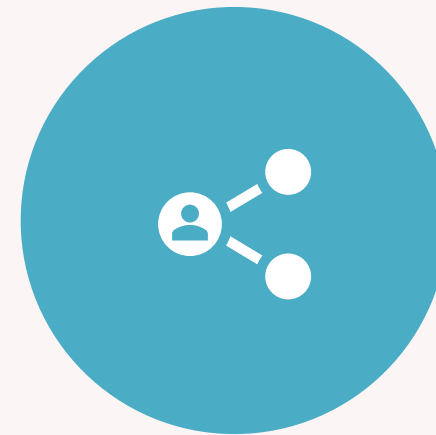
- Quality Circles: Small group problem-solving



- Quality Training: Skill enhancement programs



- Tool Implementation: Adoption of quality tools



- Best Practice Sharing: Knowledge transfer across teams

Management Review dan Reporting

- - Quality Status Reports: Regular reporting kepada management
- - Trend Analysis: Long-term quality performance
- - Improvement Recommendations: Proposed enhancements
- - Resource Allocation: Budget untuk quality initiatives



Control Quality - Overview



- DEFINISI: PROSES MEMANTAU DAN MEREKAM HASIL AKTIVITAS QUALITY CONTROL



- FOKUS: IDENTIFIKASI CAUSES OF POOR QUALITY DAN RECOMMEND ACTIONS



- OUTPUT: VERIFIED DELIVERABLES, QUALITY CONTROL MEASUREMENTS

Inspection dan Testing

- - Code Inspection: Formal review of source code
- - Unit Testing: Testing individual components/modules
- - Integration Testing: Testing interfaces antara components
- - System Testing: End-to-end testing seluruh system
- - User Acceptance Testing: Validation oleh end users



Testing Types dalam Software Development

- - Functional Testing: Verification of functional requirements
- - Non-functional Testing: Performance, security, usability
- - Regression Testing: Ensuring changes don't break existing functionality
- - Exploratory Testing: Unscripted testing berdasarkan tester knowledge

Automated Testing Strategies

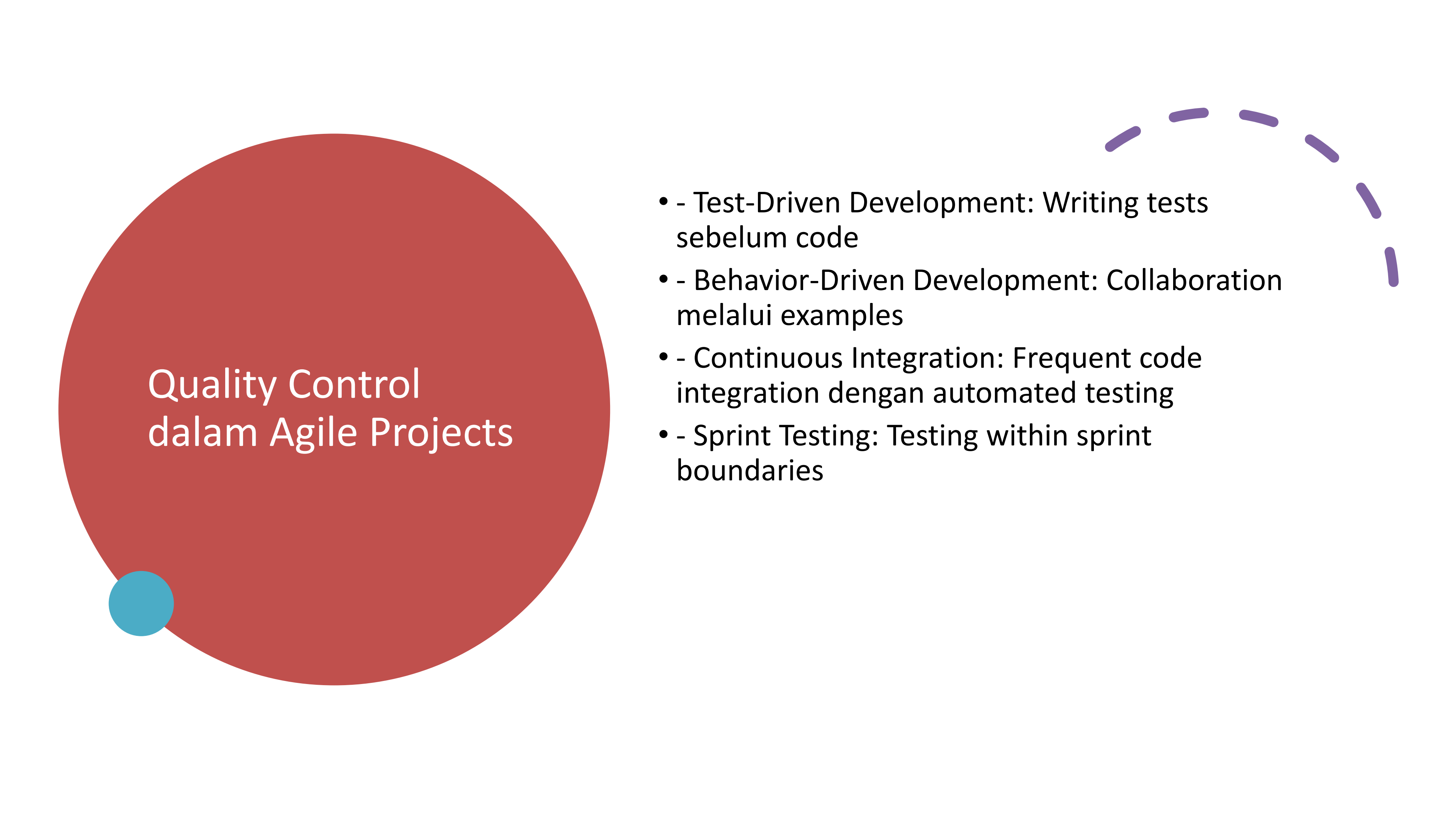
- - Test Automation Pyramid: Unit, integration, UI testing balance
- - Continuous Testing: Integration dengan CI/CD pipeline
- - Test Data Management: Preparation dan maintenance test data
- - Environment Management: Test environment configuration





Quality Control Metrics

- - Defect Density: Defects per size unit (lines of code, function points)
- - Test Coverage: Percentage of code covered oleh tests
- - Defect Removal Efficiency: Effectiveness of defect detection
- - Mean Time to Failure: Reliability measurement



Quality Control dalam Agile Projects

- - Test-Driven Development: Writing tests sebelum code
- - Behavior-Driven Development: Collaboration melalui examples
- - Continuous Integration: Frequent code integration dengan automated testing
- - Sprint Testing: Testing within sprint boundaries

Tools untuk Quality Control

- - Static Analysis Tools: Code quality analysis tanpa execution
- - Dynamic Analysis Tools: Runtime behavior analysis
- - Performance Testing Tools: Load dan stress testing
- - Security Testing Tools: Vulnerability assessment



Verification vs Validation

- - Verification: "Are we building the product right?" (Process focus)
- - Validation: "Are we building the right product?" (Product focus)
- - Contoh TI:
 - - Verification: Code review, unit testing
 - - Validation: User acceptance testing, beta testing



Configuration Management

- - Version Control: Source code management dan history
- - Build Management: Automated build processes
- - Release Management: Controlled deployment processes
- - Environment Management: Development, testing, production environments



Continuous Integration/Continuous Deployment

- - CI Practices: Frequent code integration dengan automated testing
- - CD Practices: Automated deployment pipelines
- - Quality Gates: Automated checks dalam deployment pipeline
- - Rollback Strategies: Recovery dari failed deployments



DevOps Quality Practices

- - Infrastructure as Code: Automated environment provisioning
- - Monitoring dan Alerting: Real-time system health monitoring
- - Log Management: Centralized logging dan analysis
- - Incident Management: Rapid response to quality issues



Future Trends Quality Management

- - AI dalam Testing: Intelligent test generation dan execution
- - Quality Analytics: Predictive quality forecasting
- - Shift-Right Testing: Production monitoring dan testing
- - Quality 4.0: IoT dan big data untuk quality management



DAFTAR PUSTAKA

- CMMI Institute. (2023). CMMI for Development Version 3.0.
- Crosby, P. B. (2020). Quality is Free: The Art of Making Quality Certain. Reissue Edition.
- Deming, W. E. (2018). The New Economics for Industry, Government, Education. 3rd Edition.
- ISO 9001:2015. Quality management systems — Requirements.
- ISO/IEC 25010:2011. Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE).
- ISTQB. (2023). International Software Testing Qualifications Board Syllabus.
- Juran, J. M. (2020). Juran's Quality Handbook: The Complete Guide to Performance Excellence. 7th Edition.
- Kerzner, H. (2022). Project Management: A Systems Approach to Planning, Scheduling, and Controlling. 13th Edition.
- Project Management Institute. (2021). A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition.
- Project Management Institute. (2017). A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Sixth Edition.