ASP4134 - Audit Sistem Informasi

Pertemuan III & IV:
Auditing Information Systems Acquisition & Development

Confidential. Terbatas untuk mahasiswa PKN STAN

AGENDA

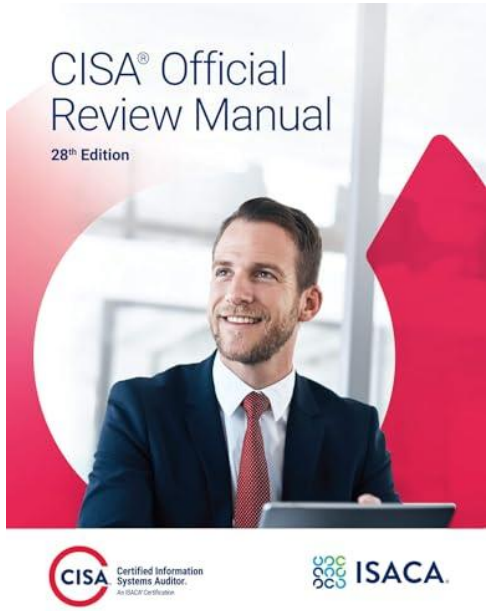
- Auditing Information Systems Acquisition & Development
- Diskusi Kelompok

ASP4134 - Audit Sistem Informasi

Key objectives:

1. memahami kerangka **konsep audit tata kelola** teknologi informasi berdasarkan standar tata Kelola
2. menyusun **program audit tata kelola** sistem informasi
3. menyusun **laporan audit tata kelola** sistem informasi
4. melaksanakan **pengauditan atas pengoperasian dan pemeliharaan** atas sistem informasi
5. melakukan **audit atas perolehan, pengembangan, dan implementasi sistem informasi**  Here we are
6. melakukan **pengauditan atas keamanan informasi**
7. menggunakan **Teknik Audit Berbantuan Komputer** dalam pengujian pengendalian dan pengujian substantif
8. menjelaskan dampak kelemahan **pengendalian umum dan pengendalian aplikasi** terhadap peluang terjadinya kecurangan dan kesalahan saji pada laporan keuangan

Referensi

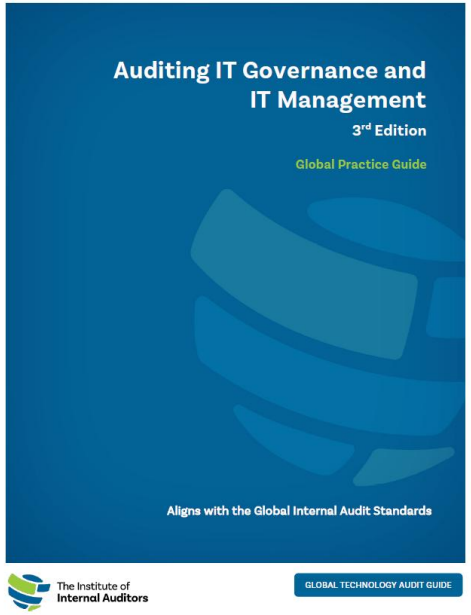


1

Useful references for this topic

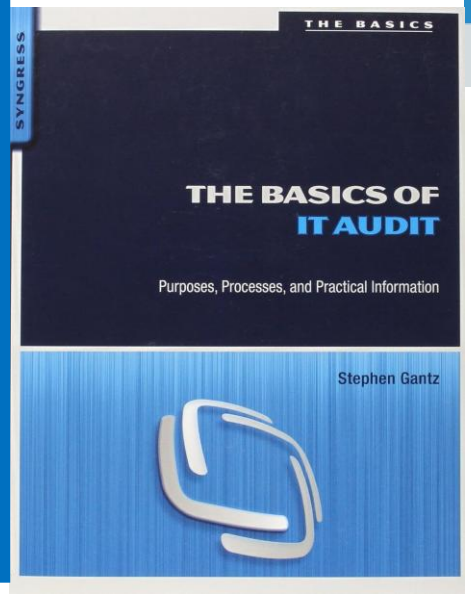
2

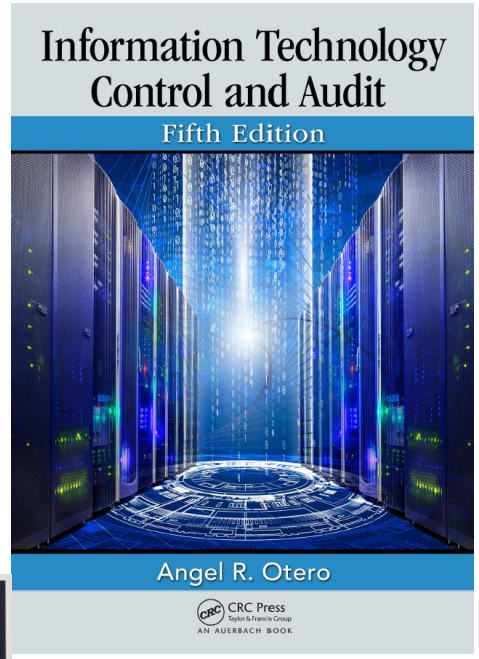




3

4





5

Last Week Recap

IT Governance = making sure IT does the right things, the right way, for the right reasons.

IT governance audit is like sending independent inspectors to check whether a city is governing its transportation system properly: ensuring direction, safety, efficiency, accountability, and performance.

Audit steps: planning – fieldwork – reporting and communication – follow up

COBIT Framework

System Acquisition & Development?

Apa itu Perolehan SI?

Proses memperoleh sistem melalui pengembangan, pembelian, outsourcing, in-house, atau cloud solution



Harus selaras dengan strategi bisnis & IT governance (COBIT 2019).

Why Auditor Needs to Audit System Acquisition & Development?

Risiko terbesar terjadi **sebelum sistem beroperasi**

salah desain = salah sistem

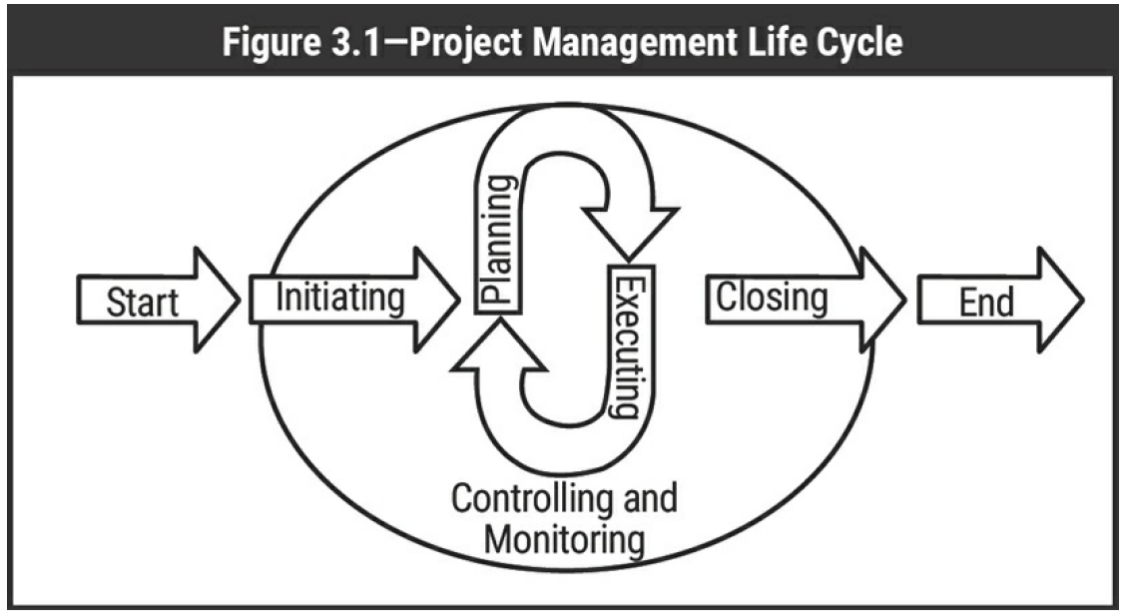
IS Auditor's role on Information System Development

- Evaluate whether the **business case** for proposed changes to information systems **meet business objectives**.
- Evaluate the organization's **project management** policies and practices.
- Evaluate **controls** at all stages of the information systems development lifecycle.
- Evaluate the **readiness** of information systems for implementation and migration into production.
- Conduct **post-implementation review** of systems to determine whether project deliverables, controls and requirements are met.
- Evaluate **change, configuration, release, and patch** management policies and practices.

PART I:
Information Systems Acquisition & Development

I.1.
Project Governance & Management

Project Management



Source: Adapted from *PMI, A Guide to the Project Management Body of Knowledge 5th Edition*, p. 50

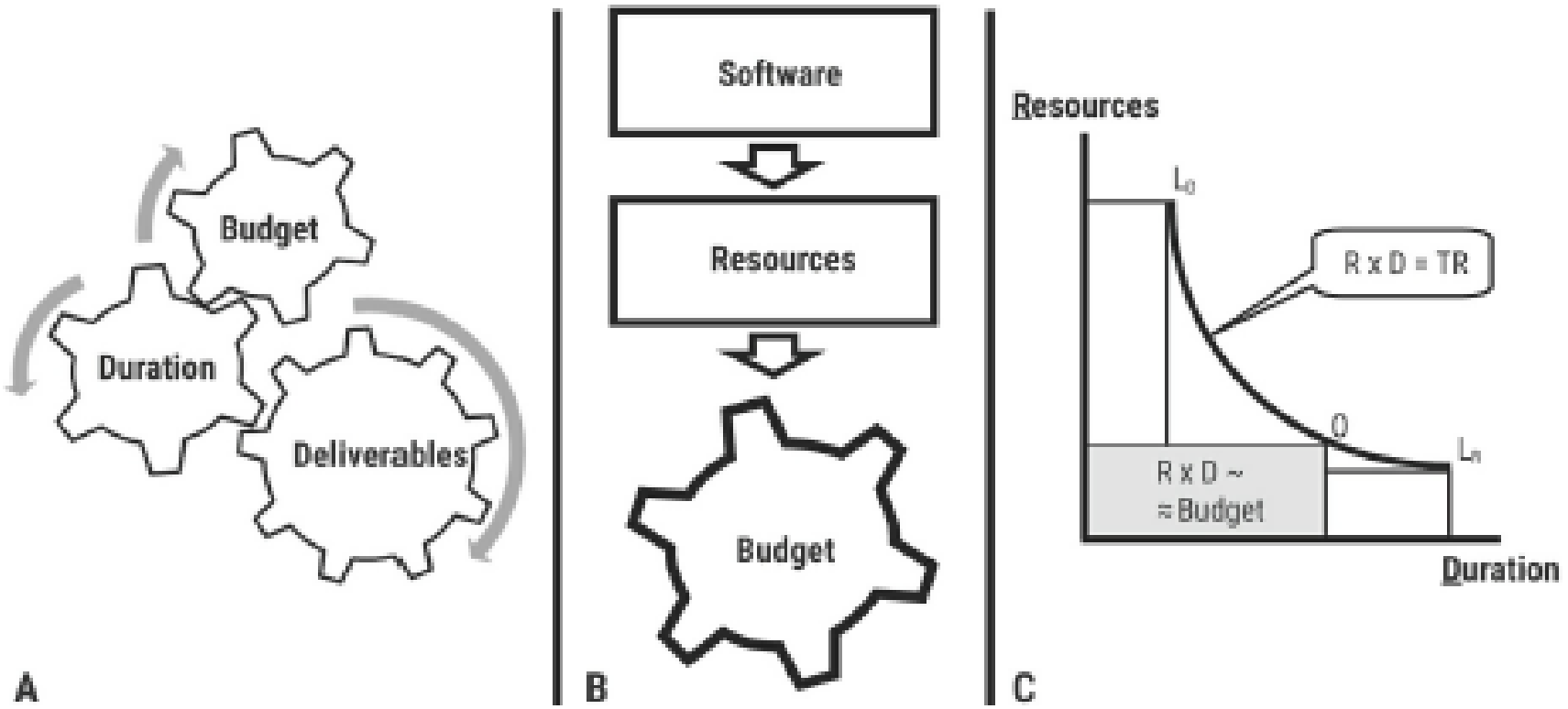
Project Management Structure

Functional Structured	Project Sturctured	Matrix structured
• No authority for project manager (PM) • PM only advises peers on project activities	• PM has formal authority on budget, schedules, and teams	• Authority is shared between PM and Dept Head

Project Roles

- Senior management
- Project sponsor
- User management
- User project team
- Project manager
- Quality assurance (QA)
- Systems development management
- Information system security engineer

Project Management Techniques



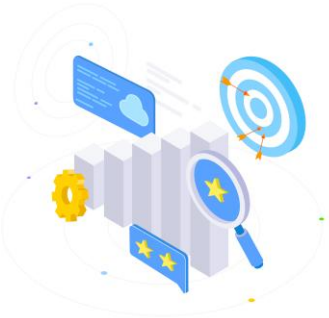
Source: Personas & Técnicas Multimedia SL © 2009

Project Estimation Methods



Project Cost Estimation

Analogous estimate – Estimates from prior projects
Parametric estimate – Past data is used to leverage statistical data
Bottom-up estimate – Detailed cost estimates of each activity
Actual costs – Extrapolation from the actual costs that were incurred on the same system during past projects



Software Size Estimation

Source Lines of Code (SLOC) – Number of lines of code
Constructive Cost Model (COCOMO) – SLOC with complexity consideration.
Function Point Analysis (FPA) – Indirect technique to estimate the size of software.

Figure 3.7—Computing Function Point Metrics

Measurement Parameter	Count	Weighting Factor			
		Simple	Average	Complex	Results
Number of user inputs		× 3	4	6	=_____
Number of user outputs		× 4	5	7	=_____
Number of user inquiries		× 3	4	6	=_____
Number of files		× 7	10	15	=_____
Number of external interfaces		× 5	7	10	=_____
Count total:					

IS Auditor's Role in

Project Management Review

The task include the following:

- To determine system objectives and requirements and to identify areas of control
- To determine **major risks** associated with the project and the relevant controls to be implemented to mitigate these
- To determine the **appropriateness** of the approach/methodology adopted for the SDLC
- To determine whether **quality assurance and UATs** are conducted before deployment of the system
- To determine the **controls to be implemented** to ensure the integrity of the production resources
- To determine the adequacy of the **post-implementation review** process

I.2.
Business Cases and Feasibility Analysis

Business Cases and Feasibility Analysis

Business cases

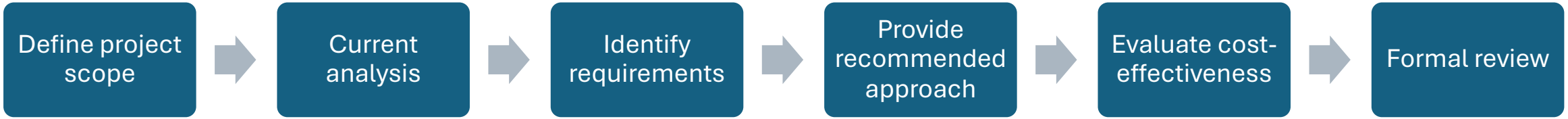
A business case is a justification for a proposed project.

"Why should this project be undertaken and/or continued?"

Feasibility Analysis

Helps to assess whether a solution is practical and achievable within the established budgets and schedule requirements.

Elements:



IS Auditor’s Role in

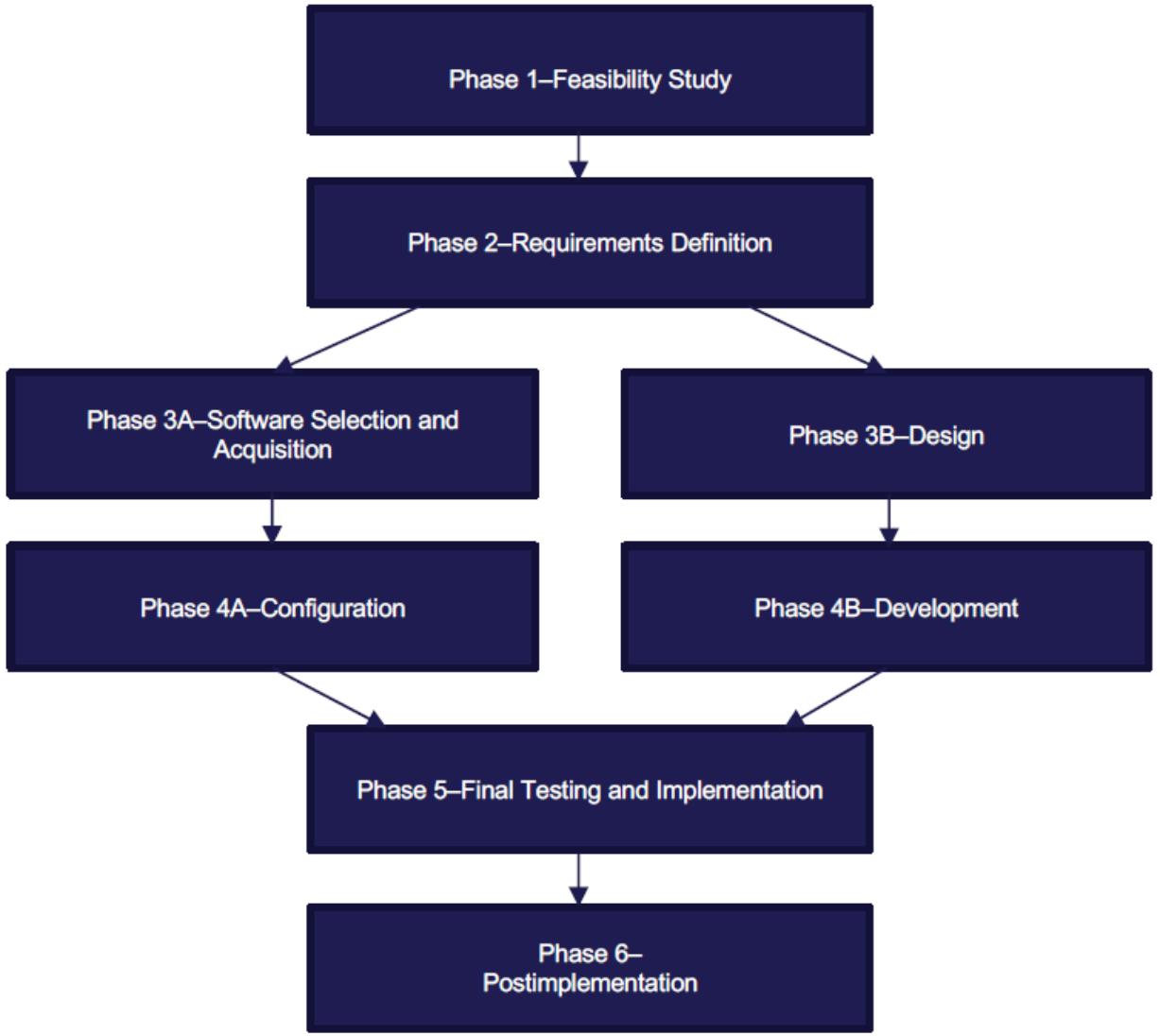
Business Cases and Feasibility Analysis

- Reviewing the **approval process** for initiation of the project.
- Reviewing the **process** followed for the development of the business case and feasibility study.
- Reviewing the process followed for the **evaluation of various alternatives.**
- Reviewing the appropriateness of the **cost justification**/benefits projected.
- Determining whether the business case and feasibility analysis are **aligned with the business objectives**.

I.3.
System Development Methodologies



Software Development Lifecycle (SDLC)



Source: ISACA, *CISA Review Manual*

DISCUSSION

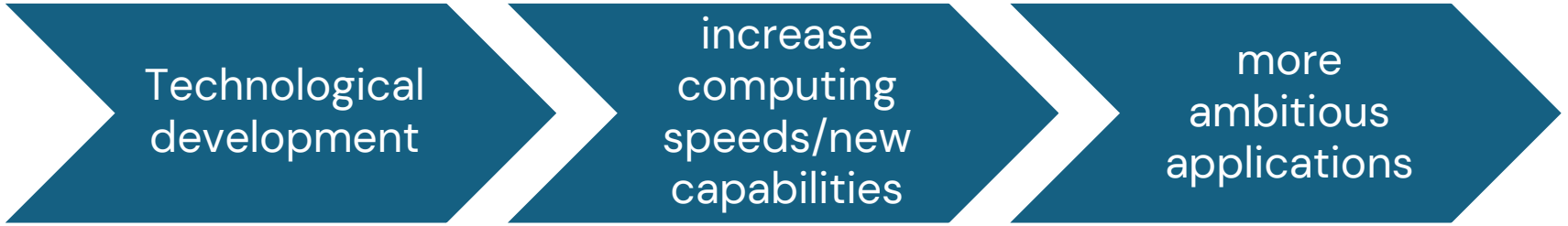
LEBIH BAIK MEMBELI ATAU MEMBANGUN SISTEM SENDIRI?

I.3.
System Development Methodologies

A. Software & Hardware Acquisition

3.A. Software Acquisition

Why Need New Software?



3.A. Software Acquisition: Key Consideration

- Business/functional/technical needs and specifications
- Cost and benefits
- Obsolescence
- Compatibility with existing systems
- Security
- Demands on existing staff
- Training and hiring requirements
- Future growth needs
- Impact on system and network performance
- Open source code versus proprietary code

3.A. Software Acquisition: Key Process

Request For Proposal (RFP)

- Product vs. system requirements
- Product scalability and interoperability
- Customer references
- Vendor viability/financial stability
- Availability of complete and reliable documentation
- Vendor support
- Source code availability
- Number of years of experience in offering the product
- A list of recent or planned enhancements to the product, with dates
- Number of client sites using the product with a list of current users
- Acceptance testing of the product

IS Auditor's Role in

Software Acquisition

- **Analyze the documentation** from the feasibility study to determine whether the decision to acquire a solution was appropriate.
- **Review the RFP** to ensure that it covers the items listed and whether the selected vendor is supported by the RFP documentation.
- **Attend agenda-based presentations** and conference room pilots to ensure that the system matches the vendor's response to the RFP.
- Review the **vendor contract** prior to its signing.
- Ensure the **contract is reviewed by legal counsel** before it is signed.
- **Review the RFP** to ensure security responses are included by the vendor

3.A. Infrastructure Acquisition

Procurement Phase

- Establish the communication process, and determine the deliverables, contracts and SLAs.
- Requirements statement is produced.

Delivery Time

- Develop delivery plan: priorities, goals, key facts, principles, communication strategies, key indicators, progress on key tasks and responsibilities.

Installation Plan

- Develop and review the plan with involved parties.

Installation Test Plan

- Develop test plan to include test cases, basic requirements specifications, definition of processes and metrics.

IS Auditor’s Role in

Infrasturcture Acquisition

- Determine if the acquisition process **began with a business need** and whether the hardware requirements for this need were considered in the specifications.
- Determine if several vendors were considered and whether the comparison between them was **done according to the aforementioned criteria.**

Risiko Pengadaan

RISIKO PENGADAAN SI	KONTROL YANG HARUS DIUJI
1. Kebutuhan tidak jelas / tidak terdokumentasi	• Ada dokumen kebutuhan (Business Requirements Document) • User & manajemen melakukan approval
2. Tender tidak transparan / konflik kepentingan	• Proses procurement mengikuti SOP • Ada panitia evaluasi independen • Rekam jejak vendor diperiksa
3. RFP/TOR tidak lengkap atau tidak sesuai kebutuhan	• Review RFP/TOR oleh user & teknis • Standar minimal spesifikasi teknis tersedia
4. Evaluasi vendor tidak objektif	• Terdapat matriks penilaian vendor • Scoring terdokumentasi & dapat ditelusuri
5. Vendor tidak kompeten / tidak berpengalaman	• Due diligence vendor • Cek portofolio, sertifikasi, keuangan, track record
6. Kontrak lemah (tanpa penalti, SLA tidak jelas)	• Kontrak mencakup SLA, KPI, penalti, timeline • Kontrak direview bagian legal & pengadaan
7. Risiko overbudget / cost overrun	• Analisis biaya: cost–benefit, total cost of ownership (TCO) • Monitoring milestone pembayaran
8. Pengamanan data & compliance tidak diatur	• Syarat keamanan (ISO/Privacy/Data Protection) ada dalam kontrak • NDA, akses data & kontrol keamanan jelas
9. Vendor lock-in (ketergantungan)	• Ketentuan transfer knowledge • Kepemilikan source code / hak lisensi diperjelas
10. SLA tidak dijalankan setelah proyek berjalan	• Monitoring SLA periodic • Laporan kinerja vendor (performance review)

AKTIVITAS KELAS: ROLEPLAYING

Proyek Pengadaan Sistem Informasi Pengelolaan Aset Daerah (SIMDA Aset)

- **Anggaran:** Rp **1.200.000.000**
- **Jenis Pengadaan:** Sistem Informasi (Software + Implementasi)
- **Metode Pengadaan:** Pemilihan langsung / tender terbatas
- Penjelasan Proyek:
  - Pemda X ingin membangun SIMDA Aset untuk mengelola inventaris aset tetap, pengadaan dan penghapusan asset, penilaian asset, integrasi dengan sistem keuangan, pelaporan aset sesuai SAP & Permendagri.
  - Saat ini semua proses aset dilakukan manual atau dengan Excel. Hal inimenyebabkan ketidaksesuaian laporan inventaris, banyak aset tidak tercatat, tidak ada audit trail, serta sulit melakukan rekonsiliasi dengan neraca keuangan. Pemda mengalokasikan Rp 1,2 miliar untuk proyek ini.

- **Peserta Roleplay:**
  - **Kelompok 1 → Management (User + Bagian Pengadaan)**
  - **Kelompok 2 → Vendor A**
  - **Kelompok 3 → Vendor B**
  - **Kelompok 4 → Auditor (Internal Audit / Inspektorat / APIP)**

AKTIVITAS KELAS: ROLEPLAYING
Proyek Pengadaan Sistem Informasi Pengelolaan Aset Daerah (SIMDA Aset)

ROLE TIAP KELOMPOK (20 Menit)

1. **Kelompok 1 → Management (User + Bagian Pengadaan)**
    1. Buatlah kriteria penilaian proyek beserta bobotnya
    2. Menyiapkan pertanyaan untul vendor
    3. Menilai vendor secara objektif dan mengumumkan pemenang
2. **Kelompok 2 & 3 → Vendor A & Vendor B**
    1. Mempresentasikan fitur, teknologi, timeline, dan harga
    2. Mempresentasikan keunggulan solusi & harga
3. **Kelompok 4 → Auditor (Internal Audit / Inspektorat / APIP)**
    1. Membuat kriteria audit
    2. Mengamati proses pemilihan vendor
    3. Memberi temuan dan rekomendasi audit

Timing

- Persiapan: 20 Menit
- Presentasi: 20 Menit (5 menit + 5 menit tanya jawab tiap kelompok)
- Management Report: 3 menit (pengumuman dan alasan kemenangan)
- Auditor’s Report: 5 menit

TUGAS PERTEMUAN III - Reflection Writing – Procurement Audit

Buatlah tulisan yang refleksikan pengalaman roleplay untuk memahami proses pengadaan IT, penilaian vendor, dan audit atas proses tersebut.

Reflection mencakup antara lain:

- Apa peran Anda dalam roleplay? Apa tugas utama kelompok Anda?
- Bagian mana dari roleplay yang menurut Anda paling menantang/berkesan?
- Risiko apa yang Anda lihat dalam proses pengadaan sistem informasi?
- Kontrol atau tata kelola apa yang seharusnya ada untuk mengurangi risiko tersebut?
- Apa pelajaran paling penting yang Anda dapatkan dari aktivitas ini?
- Siapa rekan 1 kalas yang anda rasa paling memberikan banyak insight dalam diskusi?
- Hal-hal lain yang dirasa menarik

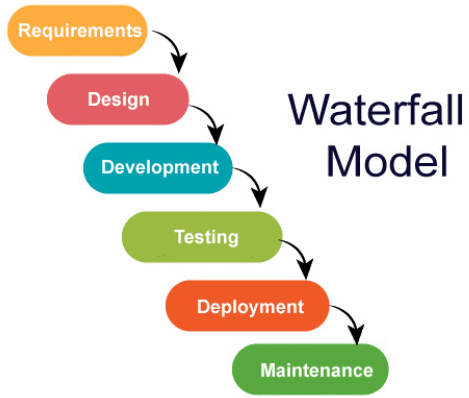
Link Pengumpulan: https://forms.gle/id6eqbgisrhaUrk98

I.3.
System Development Methodologies

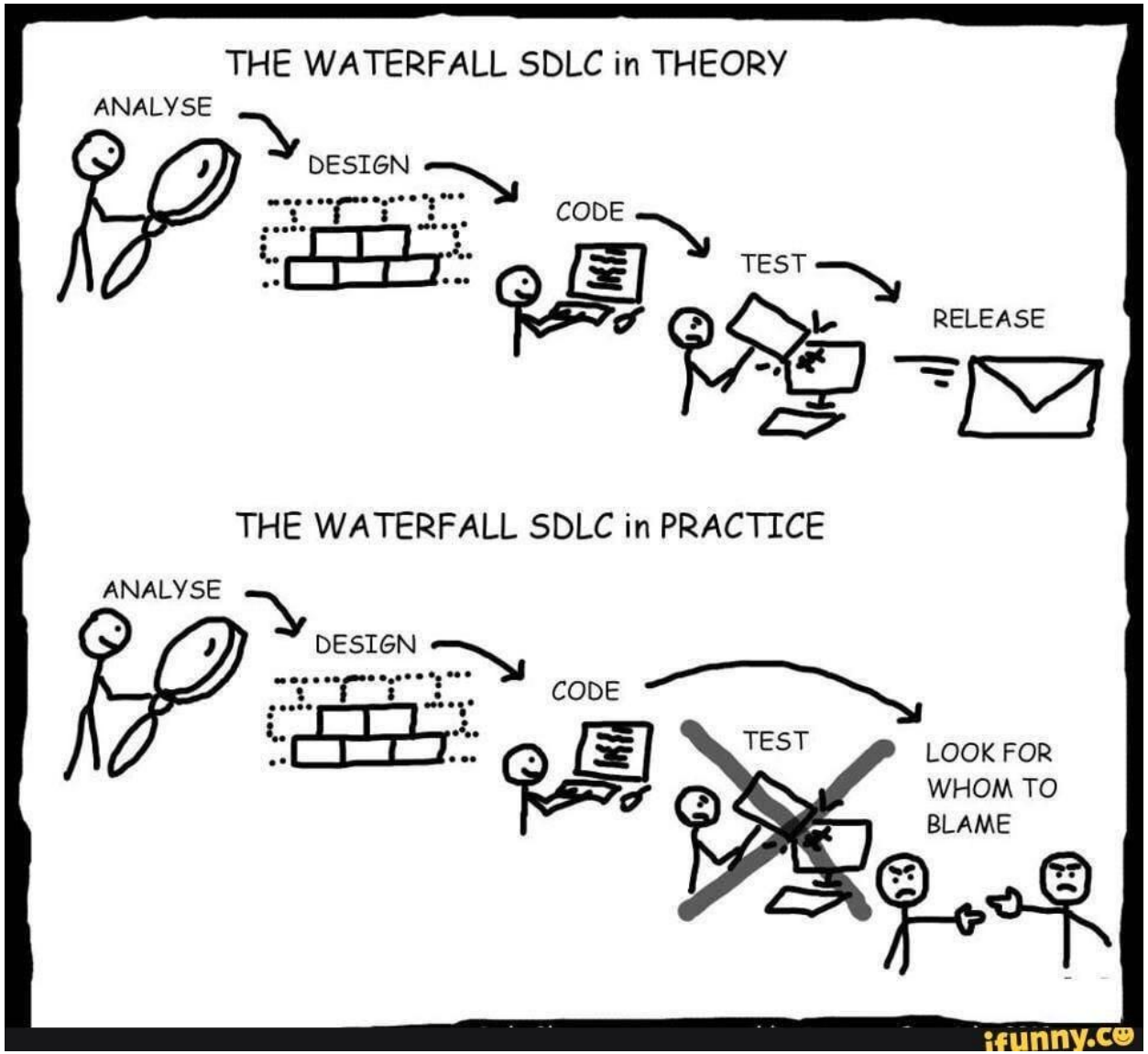
B. Software Development



3.B. Software Development Methodologies:



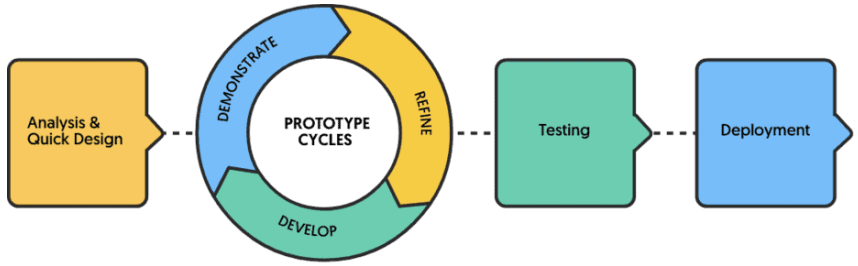
Waterfall Development
sequential development approach, in which development is seen as flowing steadily downwards (like a waterfall) through several phases



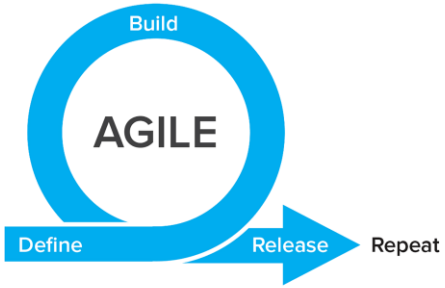
3.B. Software Development Methodologies:

Rapid Application Development

RAD relies on the prototype, which can be frequently updated to address the ever-changing user or business requirements.



A typical RAD cycle



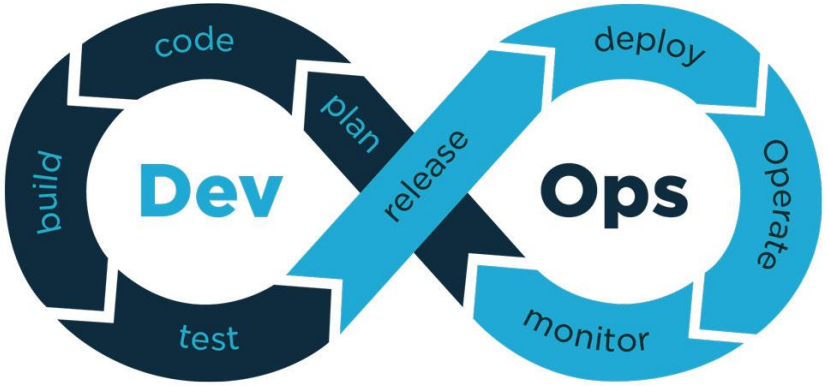
Agile Development

a group of software development frameworks based on iterative development, where requirements and solutions evolve via collaboration between self-organizing cross-functional teams.

Agile Model: Scrum, Kanban, XP, Dynamic Systems Development Method



DevOps - DevSecOps



Integration of **dev**elopment and **op**erations processes to eliminate conflicts and barriers.

An organization should consider the following **controls** when embracing a DevOps development approach:

Automated software scanning
Automated vulnerability scanning
Web application firewall
Developer application security training
Software dependency management

Access and activity logging
Documented policies and procedures
Application performance management
Asset management and inventorying
Continuous auditing and/or monitoring



DevOps - DevSecOps



Risiko Software Development

RISIKO PENGEMBANGAN	KONTROL YANG DIBUTUHKAN
1. Requirements tidak lengkap / berubah-ubah	• Formal requirements sign-off • Change Request (CR) approval & tracking
2. Tidak ada standar SDLC / dokumentasi minim	• Standard SDLC (Waterfall/Agile) diterapkan • Document repository (design, test case, API spec)
3. Developer memiliki akses ke environment produksi	• Segregation of Duties (SoD) • Access control + monitoring • Deployment via CI/CD approved pipeline
4. Tidak ada version control atau commit tracking	• Git repository mandatory • Branching strategy (main/dev/feature) • Code review sebelum merge
5. Pengujian tidak memadai (Unit/Integration/UAT)	• QA test plan • Test log & defect tracking • User Acceptance Test (UAT) wajib
6. Perubahan tidak melalui proses approval	• Formal change management • CAB (Change Advisory Board) • Impact analysis wajib
7. Poor secure coding (rentan bug & celah keamanan)	• Secure coding standard (OWASP) • Static code analysis • Peer code review
8. Tidak ada dokumentasi teknis & user	• Technical design document • User manual & onboarding

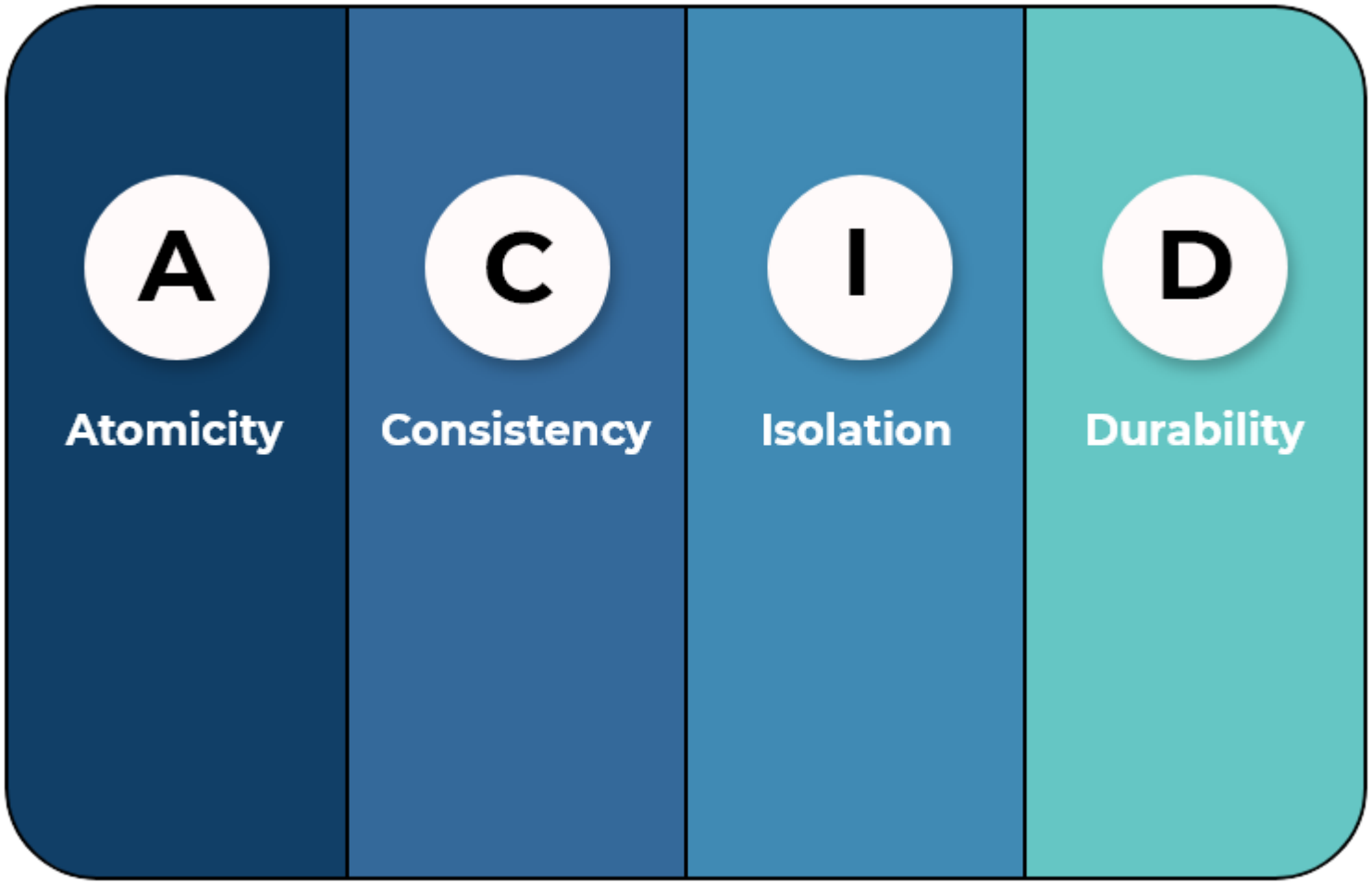
IS Auditor’s Role in

Software Acquisition

- The organization’s **change efforts are consistent with the overall culture and strategic plan** of the organization
- The reengineering team is trying to **minimize any negative impact** the change might have on the organization’s staff.
- The Business Process Reengineering (BPR) team **has documented lessons** to be learned after the completion of the BPR/process change project

I.4.
Control Identification and Design

Data Integrity Principles



Application Controls



Input

Pengendalian yang dirancang untuk memastikan bahwa data yang dimasukan ke dalam aplikasi telah diotorisasi, akurat, dan lengkap.



Proses

Pengendalian yang dirancang untuk mencegah dan mendeteksi kesalahan pada saat data transaksi diproses atau setelah data transaksi di-*entry* ke dalam aplikasi



Output

Pengendalian yang dimaksudkan untuk menetapkan bahwa data telah diproses secara lengkap dan akurat, serta informasi yang dihasikan dari aplikasi didistribusikan hanya kepada pengguna yang berhak dan tepat waktu

Types of Controls to Identify and Design

Control Type	Examples
Preventive Controls	• Access controls • Segregation of duties • Input validation • Secure coding standards
Detective Controls	• Logs • Monitoring • Alerts • Reconciliations
Corrective Controls	• Error handling • Rollback plans • Patch management

IS Auditor's Role in

Application Control Review

When reviewing application controls, an IS auditor should perform the following activities:

- Identify the **significant application components** and the **flow of transactions** through the system and gain a detailed understanding of the application by reviewing the available documentation and interviewing appropriate personnel.
- Identify the **application control** strengths and evaluate the impact of the control weaknesses.
- Develop a **testing strategy**.
- **Test the controls** to ensure their functionality and effectiveness by applying appropriate audit procedures.
- Evaluate the **control environment** by analyzing the test results and other audit evidence to determine that control objectives were achieved.
- Consider the **operational aspects of the application** to ensure its efficiency and effectiveness by comparing the system with efficient system design standards, analyzing procedures used and comparing them to management's objectives for the system.

IS Auditor's Role in

Application Control Audit

When auditing application controls, an IS auditor should perform the following activities:

- Plan the audit, set audit objectives and identify risk associated with the application being audited.
- Identify the significant application components and the flow of information through the system and gain a detailed understanding of the application by reviewing the available documentation and interviewing appropriate personnel. Developing a DFD can help visualize the flow of information.

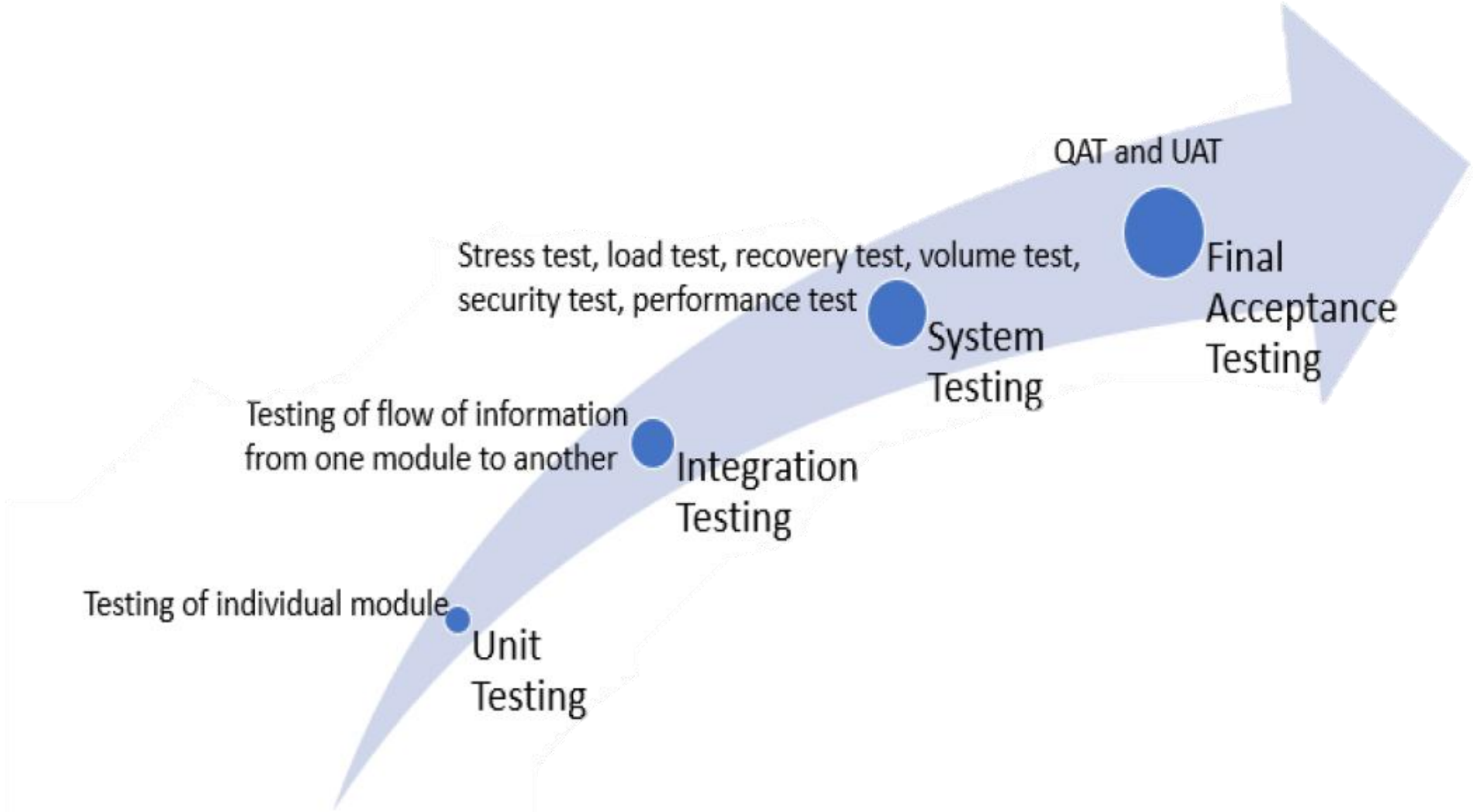
PART II:
Information System Implementation

Red Flags

- Semua serba “last minute”
- Tidak ada dry-run
- Migration tanpa testing

II.1.
System Readiness and Implementation Testing

#5. The Phases of Software Development Testing



Unit testing

- Tests program logic within a particular program or module
- Ensures that the internal operation of the program performs according to specification
- Uses a set of test cases that focus on the control structure of the procedural design

Interface or integration testing

- A hardware or software test that evaluates the connection of two or more components that pass information from one area to another

System testing

- A series of tests designed to ensure that modified programs, objects, database schema, etc., which collectively constitute a new or modified system, function properly

Final acceptance testing

- System testing that takes place during the implementation phase and applies the organization's QA methodology

Final Acceptance Testing

Quality Assurance Testing (QAT)

- Focuses on technical aspects of the application
- Verifies that the application works as documented by testing the logical design and the technology itself
- Ensures that the application meets the documented technical specifications and deliverables
- Involves minimal end-user participation
- Performed by IT department

User Acceptance Testing (UAT)

- Focuses on functional aspect of the application
- Ensures that the system is production-ready and satisfies all documented requirements
- Performed in a secure testing or staging environment that mimics production as close as possible
- Tests are written from a user's perspective
- Performed by the IT department and the end user

#2 Data Integration Testing

Data integrity testing is a set of substantive tests that examine accuracy, completeness, consistency and authorization of data presently held in a system. Two common types include:

- **Relational integrity tests**
  - Data validation routines performed at the data element and record-based levels.
- **Referential integrity tests**
  - Define existence relationships between entities in different tables of a database that needs to be maintained by the DBMS.

IS Auditor's Role in

Information System Testing (1)

- Review **the test plan for completeness**; indicate evidence of user participation, such as user development of test scenarios and/or user signoff of results; and consider rerunning critical tests.
- Reconcile **control totals** and converted data.
- Review **error reports** for their precision in recognizing erroneous data and resolution of errors.
- Verify **cyclical processing** for correctness (month-end, year-end processing, etc.).
- Verify **accuracy of critical reports** and output used by management and other stakeholders.
- Interview end users of the system for their **understanding of new methods**, procedures and operating instructions.

IS Auditor's Role in

Information System Testing (2)

- Review **system and end-user documentation** to determine its completeness and verify its accuracy during the test phase.
- Review **parallel testing results** for accuracy.
- Verify that **system security is functioning** as designed by developing and executing access tests.
- Review **unit and system test plans** to determine whether tests for internal controls are planned and performed.
- Review the **UAT** and ensure that the accepted software has been delivered to the implementation team. The vendor should not be able to replace this version.
- Review procedures used for recording and following through on **error reports**

II.2. Implementation Configuration and Release Management

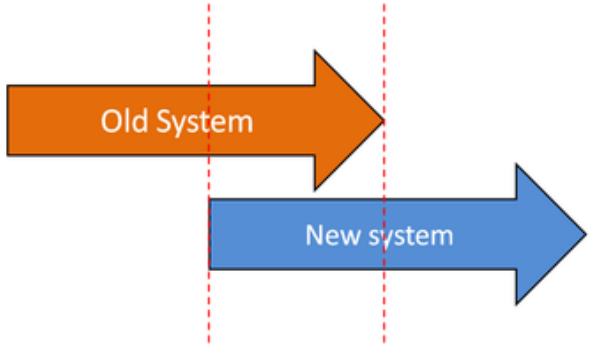


II.3
System Migration



Changeover Techniques

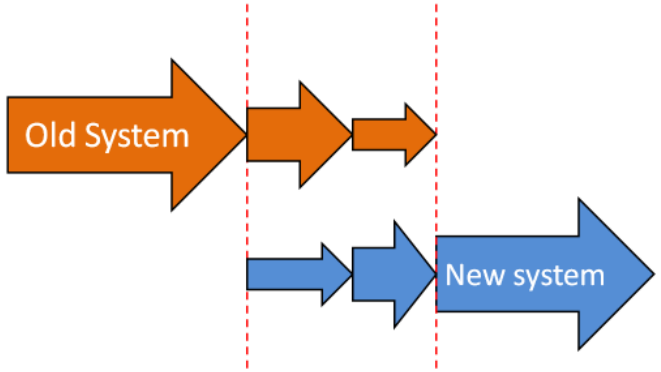
Parallel Changeover



+
Assure system meets the requirements before the old system is discontinued.

-
greatest redundancy

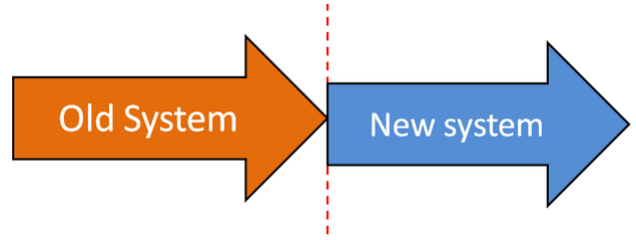
Phased Changeover



+
More time for user to adapt
Technical staff can concentrate on part of the system

-
Project duration

Abrupt Changeover/ Direct Cutover



+
Fastest
Cheapest

-
Riskiest

IS Auditor’s Role in

System Migration

IS auditors should ensure the following aspects for efficient, effective, and accurate system and data migration:

- Processes should be in place to **preserve the format, coding, structure, and integrity** of data to be migrated.
- Processes should be in place to **test the migrated data** for its integrity and completeness.
- Data conversion processes must have **audit trails and logs** to verify the accuracy and completeness of the converted data.

II.4. Post-implementation Review

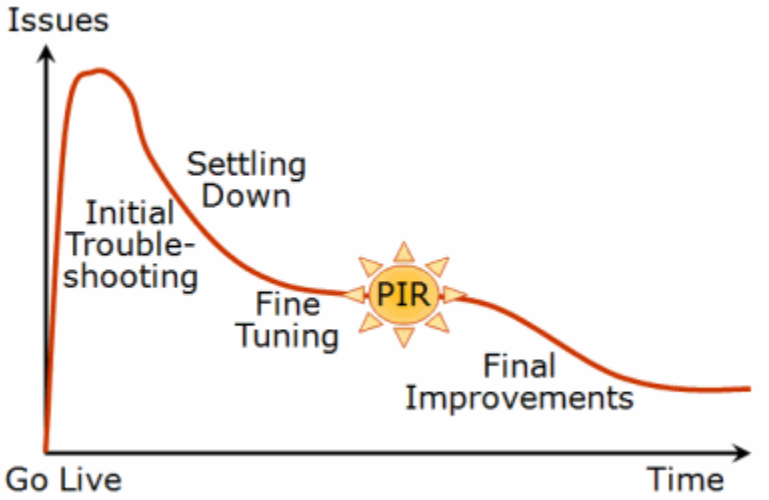
Evaluasi 30–90 hari setelah go-live.

Post-implementation Review

A post-implementation review is the process of determining and evaluating the **performance** of the system against the **requirement** and objective defined in the business case.

The project development team and business users should jointly conduct a post implementation review.

How to conduct: Interviews, Questionnaires, Feedback



IS Auditor's Role in

Post-Implementation Review

- Determine if the **system's objectives** and requirements were achieved.
- Determine the **cost-benefit analysis** and return on investment**.**
- Review **program change requests** performed to assess the type of changes required of the system.
- **Review controls built into the system to ensure that they are operating according to design.**
- Review **operators' error logs** to determine if there are any resource or operating problems inherent within the system.
- Review **input and output control balances and reports** to verify that the system is processing data accurately.

TUGAS PERTEMUAN IV

Cari satu kasus nyata (dari berita / artikel / laporan / blog / jurnal) yang berkaitan dengan:

- gagalnya pengadaan sistem informasi, atau
- masalah dalam pengembangan sistem (SDLC)

Buatlah:

1. Ringkasan Kasus (maks 5–7 kalimat)
2. Analisis Auditor (dari sisi bahasan pertemuan 3):
    1. Pada tahap apa masalah terjadi?
    2. Risiko apa yang muncul?
    3. Apa 2–3 kontrol yang seharusnya ada?
    4. Rekomendasi Audit (maks 3 poin)

Refference

- ISACA, *IT Audit Framework 4th Edition*
- ISACA, *CISA Review Manual 27th Edition*
- ISACA, *Systems Development and Project Management Audit/Assurance Program*, 2009
- ISACA, *Information Systems Auditing: Tools and Techniques: Creating Audit Programs*, 2016