

Analysis & Design — Chapter study guide

Outcome 2 — Analysis and Design · 16 key-knowledge points · exam overview, links & practice

The big picture

Analyse a need or opportunity, prepare a project plan and a software requirements specification, and produce a design for a software solution.

Unit 3 Outcome 2 covers **analysis and design**: working out what to build and planning it before any code is written. It runs from understanding why a project exists, through gathering requirements and modelling the system, to designing the data, interface and logic.

The thread: reasons/brief → plan with a Gantt chart → collect data → define requirements, constraints, scope → model with context/DFD/use case diagrams → record everything in an SRS → generate ideas (ideation) → design with tools, applying UX and UI principles → set evaluation criteria.

WHAT THE EXAM REWARDS

- Gantt-chart reading — especially dependencies, milestones and the critical path.
- Diagram components — know what belongs in a context diagram vs a Level 1 DFD, and use-case include vs extend.
- Classifying requirements and constraints from a scenario.
- Legal: match a scenario to the right Act and principle (APP/IPP); copyright is automatic.
- Backups — the full/incremental/differential comparison and restore chains.
- UX characteristics and UI design principles — be ready to evaluate/improve a given screen.

Key knowledge at a glance

KK01	Reasons why individuals and organisations undertake software development projects Why individuals/organisations start software projects.
KK02	Features of a brief that documents a problem, need or opportunity Features of a brief that documents a problem, need or opportunity.
KK03	Features of project management using Gantt charts Gantt charts: tasks, sequencing, time, dependencies, milestones, critical path, monitoring.
KK04	Methods for collecting data to determine needs and requirements Data-collection methods: interviews, observations, surveys, reports.
KK05	Characteristics of functional and non-functional requirements Functional vs non-functional requirements (analysis focus).
KK06	Constraints that influence solution development Constraints on development: economic, legal, social, technical.
KK07	Characteristics of solution scope Solution scope: version/solution boundaries.
KK08	Analytical tools for depicting the relationships between users, data and systems Analytical tools: context diagrams (L0), data flow diagrams (L1), use case diagrams.
KK09	Purpose and features of software requirements specifications Purpose and features of a software requirements specification (SRS).
KK10	Key legal requirements relating to intellectual property, ownership and privacy of data Legal requirements: Copyright Act 1968, Privacy Act 1988 (APPs), Privacy & Data Protection Act 2014 (IPPs).

KK11	File management techniques File management: naming conventions, version control, backups (full/incremental/differential), security, disposal.
KK12	Ideation techniques and tools for generating design ideas Ideation techniques: mood boards, brainstorming, mind maps, sketches, annotations.
KK13	Criteria for evaluating design ideas and the efficiency and effectiveness of solutions Criteria for evaluating design ideas and solution efficiency/effectiveness.
KK14	Design tools for generating solution designs from design ideas Design tools to generate solution designs: data dictionaries, mock-ups, object descriptions, IPO charts, pseudocode.
KK15	Characteristics of user experience (UX) and how these affect software design User experience (UX): affordance, interoperability, security, usability.
KK16	Design principles that influence the appearance and functionality of the user interface(s) UI design principles: alignment, balance, contrast, space, text formatting, usability, navigation.

Must-know glossary

Brief	A short document outlining the problem/need/opportunity, users, languages, feasibility and originality.
Critical path	The chain of dependent tasks setting the minimum project duration.
Context diagram	Level 0 model: the whole system as one process with entities and data flows.
Data flow diagram	Level 1 model adding internal processes and data stores.
SRS	A document defining requirements, constraints, scope, users, technical environment and diagrams.
Affordance	A design quality where appearance suggests how to use an element.
Differential backup	Copies all data changed since the last full backup.

Before the exam — revision checklist

- I can give reasons organisations undertake projects and the features of a brief.
- I can read a Gantt chart and identify dependencies, milestones and the critical path.
- I can choose a data-collection method and justify it.
- I can classify requirements and constraints and write a scope statement.
- I can interpret context, DFD and use-case diagrams.
- I can state the purpose and features of an SRS.
- I can apply the Copyright/Privacy/PDP Acts and key APPs/PPs.
- I can compare full/incremental/differential backups.
- I can evaluate a UI using UX characteristics and design principles.

Exam practice — questions & model answers

Q1. A council wants an app for residents to report potholes. Identify one functional and one non-functional requirement, and one legal constraint.

Identify · 3 marks

Model answer

- Functional: residents can submit a report with a location/photo (1).
- Non-functional: the app is usable on a phone / reports submit within a few seconds (1).
- Legal constraint: must comply with the Privacy Act when collecting residents' personal data (1).

Q2. State which diagram you would use to show the single overall system with its external entities, and which to show internal processes and data stores.

State · 2 marks

Model answer

- Context diagram (Level 0) for the overall system and entities (1).
- Level 1 data flow diagram for internal processes and data stores (1).

Q3. The council holds residents' personal data. Explain one legal obligation and one file-management technique that helps meet it.

Explain · 3 marks

Model answer

- Obligation: under APP 11 (Privacy Act 1988) personal data must be kept secure and destroyed when no longer needed (1).
- Technique: secure backups plus controlled access/encryption protect the data (1),
- and secure disposal removes it safely once it is no longer required (1).