

Notes	Problems with Existing Systems		
This flow chart is designed to represent the basic content of Outcome 1, and should be useful for the first NAB. You will be asked general questions relating to the process of developing software.	End Users <ul style="list-style-type: none"> • Too much paperwork • Errors in system • System too slow • Bottlenecks 	Middle Management <ul style="list-style-type: none"> • Difficulty getting reports from system • Problems interacting with other systems • Not meeting targets • Complaints from customers 	Senior Management <ul style="list-style-type: none"> • System is too labour intensive • System is too costly • System causes poor service compared to competitors

- User and Technical documentation produced
- Maintenance begins immediately after acceptance testing, with corrective maintenance. Perfective and adaptive maintenance may also occur.

Documentation and Maintenance

- Modules integrated to form sub-systems to be tested
- Sub-systems combined until whole system is built.
- Acceptance testing conducted on-site

Integration and Testing of Whole System

- Coding takes place
- Depth first or top down
- Testing is conducted on modules as they are completed, using a test-harness.

Implementation and Testing of Components

- Black-Box (functional) Testing**
- Specification used to generate test cases
 - Testing conducted by separate team, errors returned for repair
- White-Box (structural) Testing**
- Testing the logic of the code, with every logical path
 - Generally used when black-box returns an error

- Top-down process, using Structure Charts, then pseudocode, to refine the design. Data Flow may be analysed with DFDs.
- Programmers' work must integrate well
- Verification and validation occurs
- Test cases are devised and dry runs are conducted on the pseudocode.

Software Design and Validation

Software Development

Software Development Process

System Specification

- Describes the system which will be developed
- Objectives/Constraints** – aims/capacity/budget
- Hardware Specification**
- Software Specification** – Software Development Environment; Data and Functional Descriptions
- Project Issues** – cost/schedule

Analysis of Operational Requirements

- Look at ORS and model solution
- Organise development (schedule, personnel, etc)
- Produce System Specification

Technical Economic Legal Schedule

- Cheap and quick – helps management decide
- Conducted by Project Leader

Problem Definition

Perceived need within the organisation (see box)

Project Leader

Feasibility Study

Systems Analyst(s) Appointed

System Investigation

- Analyse Existing System**
- Interviews*
 - Questionnaires*
 - Observation/Task Analysis*
 - Document Analysis*
 - Background Information*
 - Specialist Knowledge*

Model New System

- Data Flow/Object*

Produce ORS

- Conducted by Systems Analyst

Functional Specification

- What it must do

Physical Specification

- What it must use

Data Requirements

- Storage space

System Prospectus

- Schedule, etc

- Offers client and consultant mutual protection
- Must be unambiguous

Operational Requirements Specification

Programmers Appointed

Personnel

- Client
- Management
- Consultants
- Project Leader
- Systems Analyst
- Programmers