What is Systems Development?

1. Finding out what needs to be done.
2. Planning what needs to be done.
3. Carrying out that plan.
4. Evaluating what has been done.

Life Cycles and Methodologies

Successful systems development will require developer(s) to follow a **systems development life cycle** (SDLC) and to use a development **methodology** (method).

There are many kinds of computer-based systems and different ways of developing them, thus a wide range of SDLCs and methodologies.


Systems Development Life Cycles

A *life cycle* covers the birth, life and death of an organism or of an ecosystem.

A *SDLC* covers “the phases a software product goes through between when it is conceived and when it is no longer available for use.”

(FOLDOC, Free On-Line Dictionary of Computing.)

“ … typically includes the following: requirements analysis, design, construction, testing (validation), installation, operation, maintenance, and retirement.”

A Systems Development Life Cycle

The classic systems development life cycle (SDLC) can be seen as having 6 (more or less) stages:

- Feasibility Study
- Systems Analysis
- Systems Design
- Programming
- Implementation
- Maintenance

Feasibility Study

A brief examination of current system, looking at alternative ways of computerising tasks.

The outcome is a *feasibility report* to management on the economic, technical and operational viability of the project.
Systems Analysis

A detailed investigation of the operations of the current system where necessary, and, more particularly, the requirements of its successor.

Models of the current system may be developed.

The outcome is a requirements specification.

Systems Design

First outline solution(s) developed, followed by a detailed specification of the required system.

Models of the required system are developed.

The outcome is detailed design documentation.

Programming

The working system is developed, ie coded, tested and documented.

The outcome is code and documentation.

Implementation

Includes the installation of hardware and/or software, user training and system changeover.

Maintenance (and Review)

Involves rectifying any problems encountered, providing support and maintenance on an ongoing basis.

As new requirements emerge, a new system may be considered, resulting in a feasibility study and the development cycle begins again.
Types of Systems Development Life Cycles

A systems development life cycle can be:

- Linear (phased)
- Iterative (evolutionary)
- Incremental (combination linear and iterative)

See the paper ‘Selecting a Development Approach’ for further details of each SDLC, plus:

- Strengths
- Weaknesses
- Situations where most appropriate
- Situations where least appropriate

BUT think life cycles wherever it refers to methodologies!

Linear or Phased Development

One type of SDLC, where the system is developed in a number of stages, is known as linear (phased) development, eg the Waterfall model.

Here a new stage is only begun when the previous stage is finished.

![Diagram of Waterfall Model]

Feasibility Study → Systems Analysis → Systems Design → Programming → Implementation → Maintenance
The Practical Waterfall Model

Feasibility Study

Systems Analysis

Systems Design

Programming

Implementation

Maintenance

Evolutionary Life Cycles

Evolutionary life cycles build the system from rough incomplete versions up to the final system.

Examples: prototyping, rapid application development (RAD) and spiral life cycles.

Prototyping

There are 2 major different approaches to the practice of prototyping. These are known as:

- Rapid prototyping
- Evolutionary prototyping
Prototyping

What is a Method (Methodology)?

“A software development method is usually based on a life cycle model of system development and has a number of development phases with a set of steps and rules for each phase. Whereas a life cycle coarsely partitions the development of a system into stages, a development method takes a life cycle and further divides each of the stages into a number of steps. A development method will prescribe in great detail what tasks are involved in each step, the nature of each task, the order in which the tasks need to be done, what documents are produced at each stage and what documents are required as input to each stage. In fact, it provides a detailed plan for producing a system.”

Methodologies

According to FOLDOC:

“1. An organised, documented set of procedures and guidelines for one or more phases of the software life cycle, such as analysis or design. Many methodologies include a diagramming notation for documenting the results of the procedure; a step-by-step ‘cookbook’ approach for carrying out the procedure; and an objective (ideally quantified) set of criteria for determining whether the results of the procedure are of acceptable quality.

2. A pretentious way of saying ‘method’.”

Examples of Methodologies

- Yourdon Structured Method
- Structured Systems Analysis and Design Method (for large scale projects)
- MERISE (for large scale projects)

Methodologies for web sites:
- December
- IBM
- Simple Web Method (SWM and SWM2)

Object-oriented methodologies:
- Booch
- Object-Oriented Hypermedia Design Method
Examples of Methodologies (contd)

RAD methodologies (Rapid Application Development)/Agile Development:

- Dynamic Systems Development Method (for large scale projects)
- Scrum (team development)
- EXTreme Programming

Refer to Gary Griffiths’ web site on the intranet.

For a database project:

- Database Design Methodology (Connolly & Begg)
According to Connolly & Begg in ‘Database Systems’, this consists of:

- Database planning
- System definition
- Requirements collection and analysis
- Database design
- DBMS selection (optional)
- Application design
- Prototyping (optional)
- Implementation
- Data conversion and loading
- Testing
- Operational maintenance

Database Design Methodology by Connolly & Begg in ‘Database Systems’

Chapter 15
Methodology – Conceptual Database Design

Chapter 16
Methodology – Logical Database Design for the Relational Model

Chapter 17
Methodology – Physical Database Design for Relational Databases

Chapter 18
Methodology – Monitoring and Tuning the Operational System
UML ( Unified Modelling Language)

UML is not a methodology. It is the international standard modelling notation used for object-oriented analysis and design.

From uml.tutorials.trireme.com:

“UML is a standardized notation for object-oriented analysis and design. However, a method is more than a notation. To be an analysis or design method it must include guidelines for using the notation and methodological principles. To be a complete software engineering method it must also include procedures for dealing with matters outside the scope of mere software development: business and requirements modelling, development process, project management, metrics, traceability techniques and reuse management.”

Choosing a Methodology

➢ Research system development life cycles.
➢ Select an appropriate system development life cycle for your project.
➢ Research systems development methodologies.
➢ Select an appropriate systems development methodology for your project.
➢ In your report, discuss the alternatives and justify your choices.
➢ Use your chosen systems development life cycle and methodology.
➢ In your report, show where and how you have used your chosen systems development life cycle and methodology.