SYSTEM DEVELOPMENT AND THE WATERFALL MODEL

What is a System? (Ch. 18)

- A system is a set of integrated components interacting with each other to serve a common purpose.
- A computer-based system is a system composed of computers and related information technology components working together to perform a specified task.

Basic Elements of a System

- A computer based system consists of the following basic elements:

  i. Environment. Anything outside the boundaries of a system.

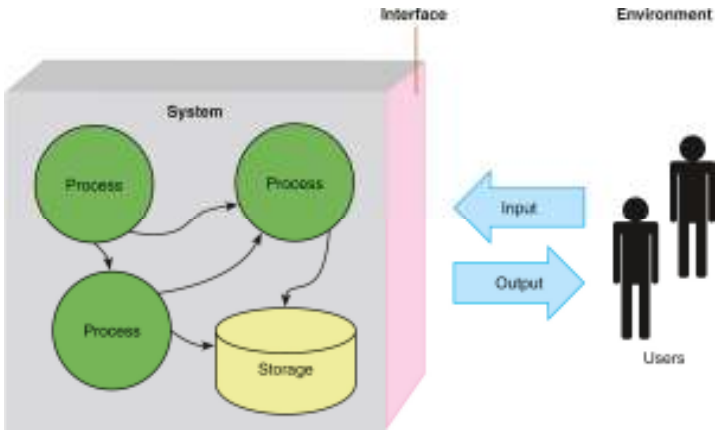
  ii. Input. Including capital, manpower, information, hardware and energy.

  iii. Output. Products and services.

  iv. Process (component). The data or information inputted into the system is managed by system **processes**.

  v. Interface. The input and output of data transferred between the system and the users take place through an **interface**.

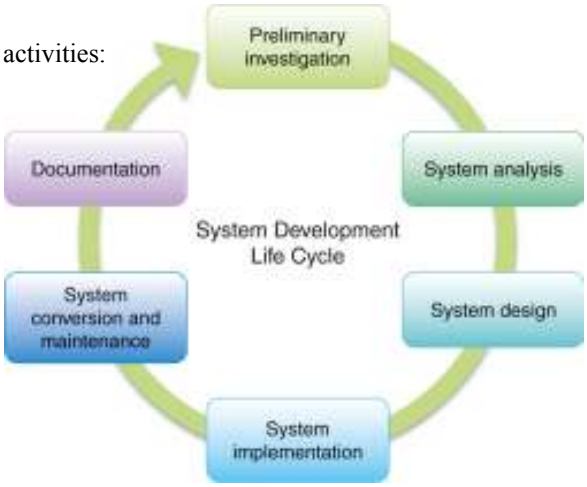
  vi. Storage. Keeping useful information during the interaction of the processes.



- For a computerized book searching system in a library, the basic elements are listed in the table below:

Environment	The library is located on the school campus.
Input	• Desktop computers, bar code readers, database server, networks • Library software • Information about books (e.g. book names, author names and keywords
Output	Search results are displayed on the computer screen.
Process	Search, borrow, return and renewal of books are performed by different processes.
Interface	The layout of the program functions is displayed on the computer screen.
Storage	The database server stores the information about books, students and circulation.

System Development Life Cycle (SDLC)



- System development can be decomposed into the following activities:

  i. Preliminary investigation

  ii. System analysis

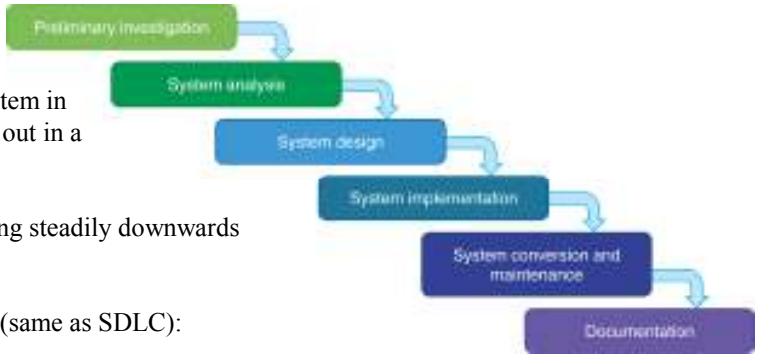
  iii. System design

  iv. System implementation

  v. System conversion and maintenance

  vi. Documentation

The Waterfall Model



- The waterfall model is the oldest and the most classical method for development a system in which the development activities are carried out in a **sequential** and **systematic** way.

- The development process is viewed as flowing steadily downwards like a waterfall.

- There are six phrases of the waterfall model (same as SDLC):

  Preliminary investigation, System analysis, System design, System implementation, System conversion and maintenance, and Documentation.

  One should only proceed to the next phase after the current phase ahs been completed and perfected.

- The strategy of the waterfall model is to try to discover bugs or errors in the system as early as possible in the development process.

- The model is also a document-driven model in which details of every aspect of the system is documented during each phase.

- Two problems:

  i. It cannot adapt well to rapid changes in user requirement.

  ii. It will take a long time for the system to become visible to the users.

System Analysis

- It consists of
  i. defining the problem,
  ii. identifying its causes,
  iii. specifying the solution and
  iv. identifying the system requirements.
- The personnel who perform system analysis are known as _____systems analyst__.

Collecting User Requirements

- Information to be collected include:
  i. the function of the system
  ii. the users of the system
  iii. major data flow / workflow of the system
  iv. the existing hardware and software of the system
  v. process time and capacity
- There are several ways to collect the above information:
  i. ***Interview***.
     a. Individuals or groups.
     b. Current users or potentials users.
     c. The best source of _____qualitative_____ information (opinions, policies and subjective description of problems)
  ii. ***Questionnaires***
     a. A good way of collecting _____quantitative_____ data of a system, such as process time and frequency of using the system.
     b. Advantage:
        - Large scale
        - Easy to analyse the result
     c. Disadvantage:
        - No observation of the reactions of the respondents.
        - High cost of designing and distributing the questionnaires.
  iii. ***Documentation Reviews***
     a. Documents include written policy manuals, regulations, standard operating procedures and program codes.
     b. They help the systems analyst understand the existing system.
  iv. ***Observation***
     a. The systems analyst can observe how documents are handled and how processes are carried out.
     b. It is the most effective way of find out the problems of a system.

Feasibility Study

- ***Technical Feasibility***.

  i. The systems analyst looks at whether the equipment, software technology and personnel involved in a solution are available.

  ii. If a new type of technology is required, the systems analyst estimates the possibility of developing it.

- ***Economic Feasibility***.

  i. The cost (building cost + running cost) and the benefits.

- ***Operational Feasibility***.

  i. We must make sure that operation of the new system is smooth and acceptable to all users.

System Requirement Specification (SRS)

- It is a document where the requirements of a system to be developed are listed. It consists of three parts:

  i. ***Business requirement.*** The requirement from the organization.

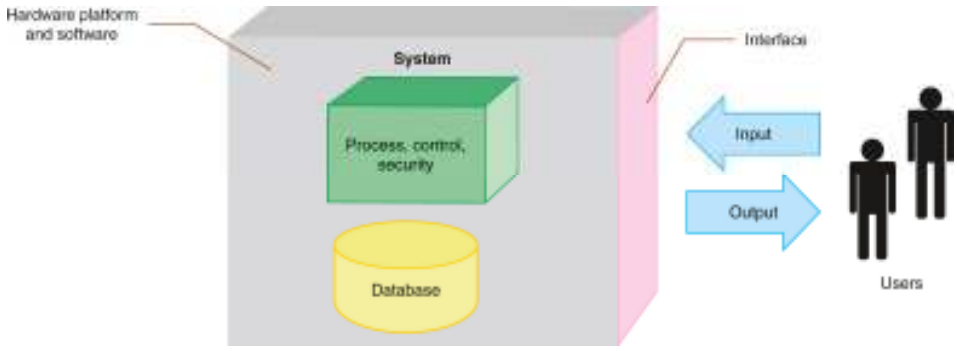
  ii. ***Functional requirement.*** The description of how the system fulfil the business requirement (input / output)

  iii. ***Non-functional requirement.*** Aspects of the system other than the specific functions including system performance, costs and security.

System Design

- The functional parts of a computer-based system:

  i. Hardware platform
  ii. Software
  iii. Input
  iv. Output
  v. User Interface
  vi. Database structure
  vii. Data control
  viii. Data security and system security



Hardware Platform

- Hardware platforms differ mainly in ___computational power___ and ___storage capacity___.
- The choice of hardware is affected by how ___advanced___ technology is at the time of making decision.
- For critical systems, supporting services such as routine and preventive maintenance are crucial to keep the business operations uninterrupted.

Software

- In general, there are three ways to acquire the software for a system:

i. Developing ___custom software___.

a. It is done by in-house programmers or systems analysts.

b. ***Advantages***:

- It caters for any special business needs.
- It is totally owned by the organization.

c. ***Disadvantages***:

- Its development time can be __long___.
- The cost and risk can be ____high___.

ii. Purchase ___commercial, off-the shelf___(COTS) software.

a. COSTS software has been tested and is available in the market.

b. ***Advantages***:

- Adopting it can reduce initial and maintenance costs.
- Reliability and functionality can be guaranteed.

c. ***Disadvantages***:

- Customization of COTS software may be limited.
- The control over the software's functionality will be lost.

iii. Employing an __application service provider_____ (ASP)

a. ***Advantages***:

- As the ASPs are specialized in developing IT applications, the software they develop can probably meet the needs of most organizations.
- There is no need for the organization to hire or train a group of IT staff.

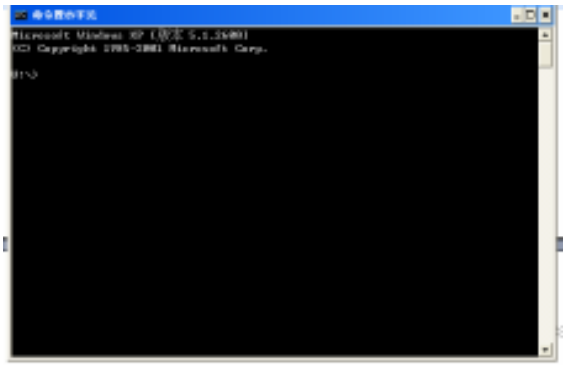
b. ***Disadvantages***:

- General control of organization data, client's privacy, information systems and project schedules may be sacrificed.
- If the ASP ceases to operate, the project will have to be suspended.

Input and Output

- Input: capital, manpower, information, hardware
- Output: information, products or services

User Interface



An example of a command line interface (CLI)

- It is a place where users interact with the system.
- Command line interfaces (CLIs)
  - i. A user needs to enter various commands via the keyboard to control the system.
  - ii. E.g. MS-DOS, UNIX
- Graphic User Interfaces (GUIs)
  - i. It allows users to input commands by selecting menu options or clicking buttons using the mouse.
  - ii. ***WIMP***: Windows, Icons, Menu, Pointers
  - iii. E.g. Microsoft Windows, Mac OS

Two principles User Interface Design:

i. ***Visibility***:
   a. *All available system control objects such as menus and buttons must be visible to user, and*
   b. *They should provide immediate feedback to indicate that they are working properly.*

ii. ***Affordance***:
   a. *The appearance of a system control object should give essential clues on its functionality.*
   b. *In a GUI, buttons afford clicking, menus afford selecting and scroll bars afford scrolling.*

- Considerations of designing the output screen:
  - i. ***Menu design***. It shows us the least number of steps required to get from one screen to another in the system.
  - ii. ***Consistent Screen Layout***. The buttons and logos on different screens should be placed in similar positions for easy navigation.
  - iii. ***User Navigation***.
    - a. In an input form, the arrangement of the input fields determines the field navigation order. (Allow using "Tab" or "Enter" to go to the next field.)
    - b. Different users may need different access rights for the same system.
  - iv. ***Heading***. The system identification and the name of the function being used are better displayed on the top of the screen so that users can know exactly where they are within the system.
  - v. ***Data field***. The input and output field must be well labelled and neatly placed on the screen.
  - vi. D***ata validation***. E.g. length check, range check, format check and type check.
  - vii. ***Error and help messages***. They may either appear at a standard location on all screens, or next to the field in error.
  - viii. ***Font, colour and style***.

Database Structure

- A database is a centralized data store that enables data to be entered, stored and updated effectively and efficiently.
- Database hierarchy: database ➔ table ➔ record/field ➔ character / byte
- A table is a collection of related records and a record is a collection of related fields about an entity such as a person or an object.
- A key is a field which can be used to identify a record, e.g. student ID number of a student.
- The system designer must define the fields required, the data type of each field, the key fields and the relationship between tables.

Data Processing

data ➔ | Data Processing | ➔ information

- The difference between data and information is the usefulness.
- In general, data processing includes data entry, tabulation, sorting, searching, modification, calculation and analysis.

Data control

- Errors can occur during data collection, processing, storage and output of information.
- Data control is the measure used to increase the correctness of data.
- It includes validation and verification.

Data and System Security

- All computer-based systems must be protected from damage and unauthorized access.
- Setting login systems, installing a firewall and anti-virus software, and backing up data regularly are common security measures.

System Implementation

- It involves software development, data conversion, system installation and testing.

Prepare computer-based system

- Quotations of hardware and software are acquired from the vendors and the most cost effective plan will then be selected.
- Before delivering the system to the user, it is important to ensure the system is running without errors and provides all the functions needed by various types of testing.

System Testing

- A system fault means that the system concerned does not comply with at least one user requirement.

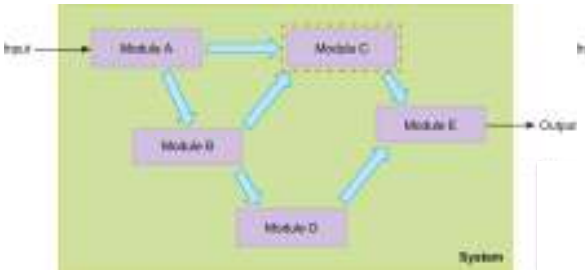
- Mistakes can occur in any stage of the SDLC.
- In order to test the system effectively and efficiently, several tests have to be done, namely a unit test, a system test and an acceptance test.

- ***Unit Test***

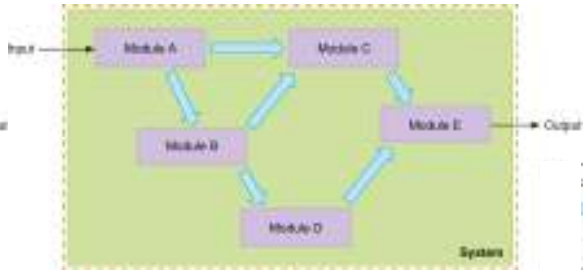
  i. A system is an integrated collection of various software **modules**. A unit test refers to testing individual modules before they are integrated with other modules.

  ii. It can be applied to **structured** or **object-oriented** software, and the unit being tested can be a function, a procedure or a method.

  iii. The testing processes are usually done by the programmers who wrote the module.



Unit Test



System Test

- ***System Test***

  i. The system test does not test individual modules but the **integration of the modules** as a complete system.

  ii. It aims to find out if the system works according to the original objectives and system requirements.

  iii. System tests are typically done by a test team who do not write the modules themselves.

  iv. Besides find system faults, a system test can be used to achieve other objects through specific tests, such as a volume test, a storage test and a performance time test:

  a. ***Volume test***. It determines whether a system can handle the volume of activities that occur during the peak of processing demands.

  b. ***Storage test***. It determines whether the storage capacity of a system is able store all transaction data.

  c. ***Performance test***. It determines the length of times used by the system to process data. It ensures the system finishes a process within a reasonable time which is usually determined by users.

- ***Acceptance Test***

  i. It aims to enable the users to determine whether the system really meets their needs and expectations. Thus, acceptance tests are written, conducted and evaluated by the users.

  ii. In an acceptance test, users prepare a set of test cases that represent typical conditions under which the system will operate when it is actually installed.

iii. The new system may be run for several days to test its daily operations as well as other functions. The opinions from users will be gathered and system developers will amend or fine-tune the system according to their reviews.

- ***Designing a test plan***

i. Each step of the testing process must planned as follows:

a. Establishing test objectives
b. Designing test cases
c. Writing test cases
d. Testing test cases
e. Executing tests
f. Evaluating test results

ii. Test objectives tell us what kind of test cases are going to be generated.

iii. Test case design is the key to successful testing. If test cases are not representative and do not thoroughly exercise the functions that demonstrate the correctness and validity of the system being tested, the testing process will become useless.

iv. Study examples on p. 165-167.

System Conversion

- System conversion is the process of replacing the existing system with a newly developed system.

- Conversion strategies:

i. ***Direct cutover conversion.***

a. The newly developed system is quickly installed, configured and made operational. The operations of the existing system are then terminated.

b. Direct cutover conversion is recommended for use if:
   - the new system is not replacing any existing system, or
   - the existing and the new system cannot be run in parallel, or
   - **downtime** of the new system can be tolerated.

ii. ***Parallel conversion***

a. Both the existing and the new systems operate for an extended period of time.

b. The extra resources required:
   - additional temporary staff
   - extra computer hardware
   - managerial and logistical complexity

c. Parallel conversion may be not possible due to:
   - data input to the existing system may not be compatible to the new one, or
   - both the new system and the existing system have to use the same hardware resources, such as database servers, or
   - the existing and the new system cannot be run in parallel by regulations, or

- the number of operational and administrative staff is sufficient to hand both system at the same time.

Advantage of using direct cutover conversion over parallel conversion:
It is simple and fewer resources are required.

Advantage of using parallel conversion over direct cutover conversion:
It provides system operation safety as there is a backup system available in case the new system fails.

iii. ***Phased conversion***

a. The new system is brought into operation through a series of phases. In each phase, one or more components of the existing system are replaced by new components.

b. Phased conversion has a lower risk than direct cutover conversion and involves less workload than parallel conversion.

c. Phased conversion adds **complexity** as dividing the conversion into phases creates more works.

d. Systems with dependent components are not suitable for phased conversion as it will be difficult to divide the conversion into independent phases.

iv. ***Pilot conversion***

a. The entire system is installed for specific sites or selected groups of such as those from certain department.

b. The system is run and tested on the pilot sites and its strengths and weaknesses and then evaluated. Any problems are corrected before the system is fully installed throughout the organization.

c. Pilot conversion has a reduced risk when compared with direct cutover conversion.

System Maintenance and User Training

- Software maintenance is the modification of a software product after delivery to
  i. correct faults;
  ii. improve performance or other attributes; and
  iii. make the product adapt to a modified environment.
- Maintenance activities include:
  i. collecting modification requests from users;
  ii. monitoring and improving system performance;
  iii. upgrading system hardware and software; and
  iv. updating system documentation to reflect system changes.
- Training provides users with hands-on experience so that they can be productive when the system is launched.
  i. Training for end users: It emphasizes the hands-on application of the system.
  ii. Training for computer operators: It can be less formal as they can self-learn.
- ***User supports*** refers to the training and assistance provided for users after the system has been launched. It is usually provided through using online documentation, on site experts or through a help desk or a forum.

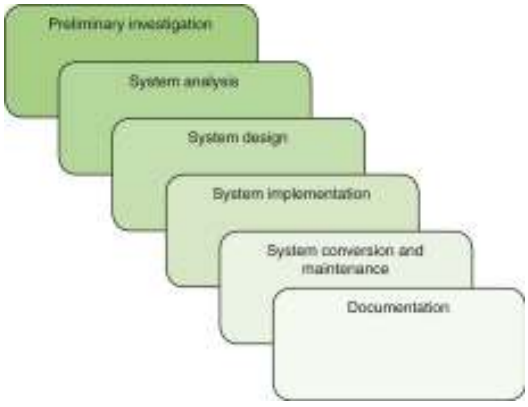
Documentation

- Documentation provides important information on how to operate and maintain a system for end users and operators respectively.
- Types of documentation include:
  - i. _____*System documentation*_____. It describes the function, architecture and construction details of a system.
  - ii. _____*Technical documentation*_____. It includes the system program source codes and comments, program flowchart, program test data produced during the system implementation stage.
  - iii. _____*User manual*____. It describes the routine operations of a system. It typical includes
    - a. System software start up and shutdown procedure
    - b. Button and keystroke sequences required to perform specific functions
    - c. Common error message
- A ***project plan*** is a formal and approved documentation to guide both project execution and project control. It includes:
  - i. Project scope
  - ii. Project time management
  - iii. Project cost and human resource management
  - iv. Project schedule

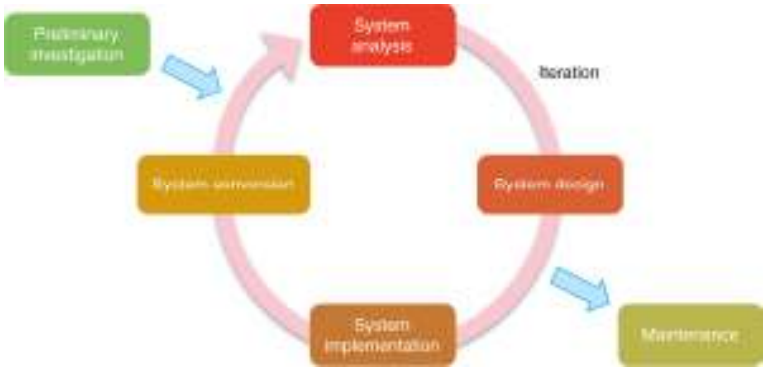
Alternative System Development Approaches (Ch. 23)

Limitations of the Waterfall Model

- TWO limitations:
  - i. The waterfall model is poorly adapted to changes in user requirements and system design.
  - ii. Design faults of a system developed by the waterfall model cannot be revealed easily until the system has been actually implemented.
- TWO variations of the waterfall model to address the above problems



Sashimi model: waterfall model with feedback



Incremental development approach: waterfall model with iteration

Prototype approach

- A ___prototype_____ is an initial, incomplete but working model of a larger and more complex system.
- The objective of building a prototype is to **collect user requirements** effectively by delivering a working system for user try-out, and it also give users a high **visibility** of the final product.

Rapid Application Development (RAD) approach

- It focuses on speeding up the system development process.
- Reason: The rapid change in the market and the development in the technological environment.
- Means to RAD approach:
  - **i.** Incremental development approach:
    - **a.** System design takes place before the analysis is completed.
    - **b.** System implementation takes place before the design is completed.
    - **c.** The cycle is then repeated with more analysis, design and implementation until the final system is completed.
  - **ii.** Prototype approach: Building working prototypes can speed up development as accurate feedback can be collected from users.
- The quality and the reliability of developed products can suffer and the cost involved may not be reduced.

The Personal

- ____*Project Manager*_____. Initiation, planning, execution and closure of a system development project
- ____*Systems Analyst*_____. Analyzing and designing systems needed by users
- ____*Programmer / Software Engineer*_____. Developing the system software in the system implementation stage
- ____*Database Administrator (DBA)*_____. Design, management and maintenance of the database
- ____*Computer Operator*_____. Carrying out scheduled system work, such as start-up, control and backup activities
- ____*Technical Support Staff*_____. Providing support for end users / Troubleshooting any problems related to the system hardware and software.