Chapter 12
Information Systems and Program Development

Objectives Overview

- Define system development and list the system development phases
- Identify the guidelines for system development
- Discuss the importance of project management, feasibility assessment, documentation, and data and information gathering techniques
- Discuss the purpose of and tasks conducted in each system development phase

See Page 510 for Detailed Objectives
Objectives Overview

- Differentiate between low-level languages and procedural languages
- Identify the benefits of object-oriented programming languages and application development tools
- List other programming languages and application development tools
- Describe various ways to develop webpages

See Page 510 for Detailed Objectives

System Development

**System development** is a set of activities used to build an information system.

System development activities are grouped into phases, and is called the **system development life cycle (SDLC)**.
System Development

System development should follow three general guidelines:

- Group activities or tasks into phases
- Involve users
- Define standards
System Development

- System development should involve representatives from each department in which the proposed system will be used.

Project management is the process of planning, scheduling, and then controlling the activities during system development.

- To plan and schedule a project efficiently, the project leader identifies the following elements:
  
  - Project scope
  - Required activities
  - Time estimates for each activity
  - Cost estimates for each activity
  - Order of activities
  - Activities that can take place at the same time
System Development

Popular tools used to plan and schedule the time relationships among project activities are Gantt and PERT charts.

System Development

- **Feasibility** is a measure of how suitable the development of a system will be to the organization.

  - Operational feasibility
  - Schedule feasibility
  - Technical feasibility
  - Economic feasibility
System Development

- **Documentation** is the collection and summarization of data, information, and deliverables.
- Maintaining up-to-date documentation should be an ongoing part of system development.

System Development

- During system development, members of the project team gather data and information using several techniques
  - Review documentation
  - Observe
  - Survey
  - Interview
  - JAD Sessions
  - Research
System Development

- The **planning phase** for a project begins when the steering committee receives a project request
- Four major activities are performed:

  - Review and approve the project requests
  - Prioritize the project requests
  - Allocate resources
  - Form a project development team

System Development

- The **analysis phase** consists of two major activities:

  - **Conduct a preliminary investigation**
    - Determines and defines the exact nature of the problem or improvement
    - Interview the user who submitted the request
  - **Perform detailed analysis**
    - Study how the current system works
    - Determine the users’ wants, needs, and requirements
    - Recommend a solution
• The system proposal assesses the feasibility of each alternative solution
• The steering committee discusses the system proposal and decides which alternative to pursue

- Modify existing system
- Buy retail software
- Use web apps
- Build custom software
- Outsource
System Development

• The **design phase** consists of two major activities

  - Acquire hardware and software
  - Develop all of the details of the new or modified information system

System Development

• To acquire the necessary hardware and software:

  - Identify technical specifications
  - Solicit vendor proposals
  - Test and evaluate vendor proposals
  - Make a decision
System Development

- The next step is to develop detailed design specifications

- Systems analysts typically develop two types of designs for each input and output
A prototype (proof of concept) is a working model of the proposed system’s essential functionality:

- Prototypes have inadequate or missing documentation
- Users tend to embrace the prototype as a final system
- Should not eliminate or replace activities

Computer-aided software engineering (CASE) tools are designed to support one or more activities of system development.
The purpose of the implementation phase is to construct the new or modified system and then deliver it to users.

Various tests should be performed on the new system:

- **Unit test**: Verifies that each individual program or object works by itself.
- **Systems test**: Verifies that all programs in an application work together properly.
- **Integration test**: Verifies that an application works with other applications.
- **Acceptance test**: Checks the new system to ensure that it works with actual data.
System Development

- **Training** involves showing users exactly how they will use the new hardware and software in the system
  - One-on-one sessions
  - Classroom-style lectures
  - Web-based training

System Development

- One or more of four conversion strategies can be used to change from the old system to the new system
  - Direct conversion
  - Parallel conversion
  - Phased conversion
  - Pilot conversion
System Development

• The purpose of the support and security phase is to provide ongoing assistance for an information system and its users after the system is implemented

  ![](perform_maintenance_activities.png)
  ![](monitor_system_performance.png)
  ![](assess_system_security.png)

Application Development Languages and Tools

• A programming language is a set of words, abbreviations, and symbols that enable a software developer to communicate instructions to a computer or mobile device
  – Low-level language
  – High-level language
Application Development Languages and Tools

- **Machine language** is the first generation of programming languages.
- Only language the computer directly recognizes.

![Machine Language Code Example]

Application Development Languages and Tools

- **Assembly language** is the second generation of programming languages.
- Programmer writes instructions using symbolic instruction codes.
- A source program contains the language instructions, or code, to be converted into machine language.
Application Development Languages and Tools

• In a **procedural language**, the programmer writes instructions that tell the computer what to accomplish and how to do it

• The **C** programming language is used to write many of today’s programs

```c
// Compute Regular Time Pay
rt_pay = rt_hrs * pay_rate;

// Compute Overtime Pay
if (ot_hrs > 0)
  ot_pay = ot_hrs * 1.5 * pay_rate;
else
  ot_pay = 0;

// Compute Gross Pay
gross = rt_pay + ot_pay;

// Display Gross Pay
printf("The gross pay is \$%d\n", gross);
```
Application Development Languages and Tools

A compiler translates an entire program before executing it.

An interpreter converts and executes one code statement at a time.
Application Development Languages and Tools

• An **object-oriented programming (OOP) language** allows programmers the ability to reuse and modify existing objects

• Other advantages include:

  - Objects can be reused
  - Programmers create applications faster
  - Most object-oriented application development tools are IDEs

Application Development Languages and Tools

• **Java** is an object-oriented programming language developed by Sun Microsystems

• The Just-in-time (JIT) compiler to convert the machine-independent code into machine-dependent code
Application Development Languages and Tools

• C++ is an extension of the C programming language
  – Additional features for working with objects
• Visual Studio is Microsoft’s suite of object-oriented application development tools that assists software developers in building programs and apps for Windows or any operating system that supports the Microsoft .NET Framework.
Application Development Languages and Tools

• A 4GL (fourth-generation language) is a nonprocedural language that enables users and programmers to access data in a database
  – One popular 4GL is SQL

Application Development Languages and Tools

• Classic programming languages include:
  - Ada
  - ALGOL
  - APL
  - BASIC
  - COBOL
  - Forth
  - FORTRAN
  - HyperTalk
  - LISP
  - Logo
  - Modula-2
  - Pascal
  - PILOT
  - PL/1
  - Prolog
  - RPG
  - Smalltalk
Application Development Languages and Tools

- An application generator is a program that creates source code or machine code from a specification of the required functionality
  - Often bundled as part of a DBMS
- A **macro** is a series of statements that instructs an application how to complete a task
- You usually create the macro in one of two ways:
  - Record the macro with a macro recorder
  - Write the macro

Figure 12-19
Application Development Languages and Tools

• **HTML** is a special formatting language that programmers use to format documents for display on the web.

  ![HTML example](image)

Application Development Languages and Tools

• **XML** allows web developers to create tags that describe how information is displayed
  – WML is a subset of XML and is used to design pages specifically for microbrowsers.
Application Development Languages and Tools

- Software developers write scripts, applets, servlets, or ActiveX controls using a variety of languages

JavaScript  Perl  PHP

Application Development Languages and Tools

Ruby on Rails provides technologies for developing object-oriented, database-driven websites
Summary

- System development phases
- Guidelines for system development
- Activities that occur during system development
- Various programming languages and program development tools
- Web development tools

Discovering Computers
Technology in a World of Computers, Mobile Devices, and the Internet

Chapter 12
Information Systems and Program Development
Chapter 12 Complete