Chapter 6
Prototyping, RAD, and Extreme Programming

Systems Analysis and Design

Major Topics

• Prototyping
• Rapid application development (RAD)
• Extreme Programming (XP)

Prototyping

• Prototyping is an information-gathering technique.
• Prototypes are useful in seeking user reactions, suggestions, innovations, and revision plans.
• Prototyping may be used as an alternative to the systems development life cycle.

Four Kinds of Prototypes

The four conceptions of prototypes are:
• Patched-up prototype.
• Nonoperational scale model.
• First-of-a-series.
• Prototype that contains only some of the essential system features.

Patched-up Prototype

• This is a working model with all the features but is inefficient.
• Users can interact with the system.
• Storage and retrieval of data may be inefficient.
• May contain only basic features.

Nonoperational Scale Models

• A nonoperational scale model is one that is not operational, except for certain features to be tested.
• Prototype input and output.
First-of-a-Series Prototype

- Pilot system is created.
- Prototype is an operation model.
- Useful when many installations of the same information system are planned.
- An example is a system to be installed in one location, tested and modified as necessary, and later implemented in other locations.

Selected Features Prototype

- An operational model includes some, but not all, of the final system features.
- With the acceptance of these features, later essential features are added.
- Some menu items are available.
- System is built in modules.
- These are part of the actual system.

Prototyping As an Alternative to the Systems Life Cycle

- Two main problems with the SDLC:
  - Extended time required to go through the development life cycle.
  - User requirements change over time.
  - Prototyping may be used as an alternative.

Prototype Development Guidelines

Guidelines for developing a prototype are:
- Work in manageable modules.
- Build the prototype rapidly.
- Modify the prototype in successive iterations.
- Stress the user interface.

Prototype Disadvantages

- Managing the prototyping process is difficult because of its rapid, iterative nature.
- Incomplete prototypes may be regarded as complete systems.

Prototype Advantages

- Potential for changing the system early in its development
- Opportunity to stop development on an unworkable system
- Possibility of developing a system that closely addresses users needs and expectations
Prototype Evaluation – The User’s Role

• Three ways the user is involved:
  • Experimenting with the prototype.
  • Giving open reactions to the prototype.
  • Suggesting additions to and/or deletions from the prototype.

Rapid Application Development (RAD)

RAD, or rapid application development, is an object-oriented approach to systems development that includes a method of development as well as software tools.

RAD Phases

• The three broad phases to RAD are:
  • Requirements planning.
  • RAD design workshop.
  • Implementation.

Requirements Planning Phase

• Users and analysts meet to identify objectives of the application or system
• Oriented toward solving business problems

RAD Design Workshop

• Design and refine phase.
• Use group decision support systems to help users agree on designs.
• Programmers and analysts can build and show visual representations of the designs and workflow to users.
• Users respond to actual working prototypes.
• Analysts refine designed modules based on user responses.

Implementation Phase

• As the systems are built and refined, the new systems or partial systems are tested and introduced to the organization.
• When creating new systems, there is no need to run old systems in parallel.
Martin Approach to RAD

The Martin approach to RAD includes four phases:
• Requirements planning.
• User design.
• Construction.
• Cutover.

RAD and the SDLC

• RAD tools are used to generate screens and exhibit the overall flow of the application.
• Users approve the design and sign off on the visual model.
• Implementation is less stressful because users helped to design the business aspects of the system.

When to Use RAD

RAD is used when:
• The team includes programmers and analysts who are experienced with it.
• There are pressing reasons for speeding up application development.
• The project involves a novel ecommerce application and needs quick results.
• Users are sophisticated and highly engaged with the goals of the company.

Disadvantages of RAD

• May try and hurry the project too much
• Loosely documented

Extreme Programming (XP)

Extreme programming (XP) takes good systems development practices to the extreme.
Four Values of Extreme Programming

The four values of extreme programming are:
• Communication.
• Simplicity.
• Feedback.
• Courage.

Five XP Principles

The five XP principles are:
• Providing rapid feedback.
• Assuming simplicity.
• Changing incrementally.
• Embracing change.
• Encouraging quality work.

Four Basic Activities of XP

The four basic activities of XP are:
• Coding.
• Testing.
• Listening, to the programming partner and customer.
• Designing.

Four XP Resource Control Variables

The four resource control variables in XP are:
• Time.
• Cost.
• Scope.
• Quality.

Four XP Core Practices

The four XP core practices are:
• Short releases, work with the most important features first.
• Having a 40-hour work week.
• Having an onsite customer.
• Pair programming with another programmer.

Agile Modeling

• Agile modeling is similar to XP.
• In addition to the values of communication, simplicity feedback and courage, has a fifth value of humility.
Agile Modeling (Continued)

- Agile modeling process is:
  - Listen to user stories.
  - Draw a logical workflow model.
  - Create new user stories based on the workflow.
  - Develop some prototypes.
  - Use feedback from the prototypes and logical workflow to create physical model.

Scrum

- Scrum is an Agile approach that has an emphasis on teamwork.
- Team success is of primary importance.
- Individual success is secondary.
- The team works within a strict time frame.
- The project leader has some but not much influence on detail.