

# SYSTEMS ANALYSIS AND DESIGN

**Implementation, Maintenance &  
Methodologies**



**CENTRAL  
UNIVERSITY**

FAITH • INTEGRITY • EXCELLENCE

Presented by  
**Dr. K. Kissi Mireku**

# Presentation Content

## ➤ *Implementation Phase*

- *Testing*
- *Training*
- *Conversion*

## ➤ Maintenance Phase

## ➤ *ISDL Methodologies*

- *Waterfall Model*
- *Iterative Model*
- *Spiral Model*
- *V-Model*
- *Big Bang Model*



# System Design final phase

## □ Testing

- Testing is the process or activity that checks the functionality and correctness of software according to specified user requirements in order to improve the quality and reliability of system.
- *It is an expensive, time consuming, and critical approach in system development which requires proper planning of overall testing process.*
- A successful test is one that finds the errors.
  - *It executes the program with explicit intention of finding error, i.e., making the program fail.*
- These errors are of two types: syntax errors and logic errors.
  - **Syntax** errors (e.g., a misspelled word or a misplaced comma) are easier to find and will not permit the program to run.
  - **Logic** errors permit the program to run, but result in incorrect output. Logic errors are more difficult to detect, because the cause is not obvious.



# System Design phase - Testing

## □ Stages of System Testing

- **Test Strategy**

- *It is a statement that provides information about the various levels, methods, tools, and techniques used for testing the system. It should satisfy all the needs of an organization.*

- **Test Plan**

- *It provides a plan for testing the system and verifies that the system under testing fulfils all the design and functional specifications.*

- **Test Procedures**

- *It consists of the steps that should be followed to execute each of the test cases. These procedures are specified in a separate document called test procedure specification.*

- **Test Result Documentation**

- *Test result file contains brief information about the total number of test cases executed, the number of errors, and nature of errors<sup>4</sup>. These results are then assessed against criteria in the test specification to determine the overall outcome of the test.*



# System Design phase - Testing

## □Types of Testing

Testing can be of various types and different types of tests are conducted depending on the kind of bugs one seeks to discover –

- **Unit Testing**

- *Also known as Program Testing, it is a type of testing where the analyst tests or focuses on each program or module independently. It is carried out with the intention of executing each statement of the module at least once.*

- **Integration Testing**

- *the analyst tests multiple module working together. It is used to find discrepancies between the system and its original objective, current specifications, and systems documentation.*

- **Functional Testing**

- *Function testing determines whether the system is functioning correctly according to its specifications and relevant standards documentation. It typically starts with the implementation of the system, which is very critical for the success of the system.*



# Implementation phase

- Implementation is the process of converting from the old system to the new system.
  - *It includes training the users, providing documentation, and conversion from any previous system to the new system.*
- Implementation is a process of ensuring that the information system is operational.
  - *It involves –*
    - *Constructing a new system from scratch*
    - *Constructing a new system from the existing one.*
- Implementation allows the users to take over its operation for use and evaluation.
- It involves training the users to handle the system and plan for a smooth conversion.



# Implementation phase

## □ Training

- The personnel in the system must know in detail what their roles will be, how they can use the system, and what the system will or will not do. The success or failure of well designed and technically elegant systems can depend on the way they are operated and used.
- **Training Systems Operators**
  - Systems operators must be trained properly such that they can handle all possible operations, both routine and extraordinary. The operators should be trained in what common malfunctions may occur, how to recognize them, and what steps to take when they come.
  - Training involves creating troubleshooting lists to identify possible problems and remedies for them, as well as the names and telephone numbers of individuals to contact when unexpected or unusual problems arise.
  - Training also involves familiarization with run procedures, which involves working through the sequence of activities needed to use a new system.



# Implementation phase

- **User Training**

- *End-user training is an important part of the computer-based information system development, which must be provided to employees to enable them to do their own problem solving.*
- *User training involves how to operate the equipment, troubleshooting the system problem, determining whether a problem that arose is caused by the equipment or software.*
- *Most user training deals with the operation of the system itself. The training courses must be designed to help the user with fast mobilization for the organization.*

- **Training Guidelines**

- *Establishing measurable objectives*
- *Using appropriate training methods*
- *Selecting suitable training sites*
- *Employing understandable training materials*





# Implementation phase

## □ Training Methods

### ❖ Instructor-led training

- It involves both trainers and trainees, who have to meet at the same time, but not necessarily at the same place. The training session could be one-on-one or collaborative.
- It is of two types –
  - **Virtual Classroom**
    - *In this training, trainers must meet the trainees at the same time, but are not required to be at the same place. The primary tools used here are: video conferencing, text based Internet relay chat tools, or virtual reality packages, etc.*
  - **Normal Classroom**
    - *The trainers must meet the trainees at the same time and at the same place. Their primary tools used here are blackboard<sup>9</sup>, overhead projectors, LCD projector, etc.*

# Implementation phase

## □ Training Methods

### ❖ Self-Paced Training

- It involves both trainers and trainees, who do not need to meet at the same place or at the same time. The trainees learn the skills themselves by accessing the courses at their own convenience. It is of two types –

- **Multimedia Training**

- *In this training, courses are presented in multimedia format and stored on CD-ROM. It minimizes the cost in developing an in-house training course without assistance from external programmers.*

- **Web-based Training**

- *In this training, courses are often presented in hyper media format and developed to support internet and intranet. It provides just-in-time training for end users and allow organization to tailor training requirements.*



# Implementation phase

## ❑ Conversion

- It is a process of migrating from the old system to the new one. It provides understandable and structured approach to improve the communication between management and project team.

## ❑ Conversion Methods

- This involves four major conversion methods:  
*parallel, direct, pilot, and phased.*

### ❖ **Parallel**

- *In a parallel conversion process, the old system and the new system operate simultaneously for a period of time. That is, both systems process the same data at the same time, and the outputs are compared.*
- *Most large systems have a parallel conversion process to lessen the risk.*



# Implementation - Conversion

- In a **direct** conversion process, the old system is cut off and the new system is turned on at a certain point in time. This type of conversion is the least expensive, but the most risky if the new system doesn't work as planned.
- The **pilot** conversion process introduces the new system in one part of the organization, such as in one plant or in one functional area. The new system runs for a period of time and is assessed. After the new system works properly, it is introduced in other parts of the organization.
- The **phased** conversion process introduces components of the new system, such as individual modules, in stages. Each module is assessed, and, when it works properly, other modules are introduced, until the entire new system is operational.



Method	Description	Advantages	Disadvantages
Parallel Conversion	Old and new systems are used simultaneously.	Provides fallback when new system fails. Offers greatest security and ultimately testing of new system.	Causes cost overruns. New system may not get fair trail.
Direct Conversion	New system is implemented and old system is replaced completely.	Forces users to make new system work Immediate benefit from new methods and control.	No fall back if problems arise with new system Requires most careful planning
Pilot Approach	Supports phased approach that gradually implement system across all users	Allows training and installation without unnecessary use of resources. Avoid large contingencies from risk management.	A long term phase in causes a problem of whether conversion goes well or not.
Phase-In Method	Working version of system implemented in one part of organization based on feedback, it is installed throughout the organization all alone or stage by stage.	Provides experience and line test before implementation When preferred new system involves new technology or drastic changes in performance.	Gives impression that old system is erroneous and it is not reliable.

# System Maintenance / Enhancement

❑ After conversion, the new system will operate for a period of time, until (like the old system it replaced) it no longer meets its objectives.

- *Once the new system's operations are stabilized, audits are performed during operation to assess the system's capabilities and determine if it is being used correctly.*

• **Maintenance** means restoring something to its original conditions.

• **Enhancement** means adding, modifying the code to support the changes in the user specification.

- *System maintenance conforms the system to its original requirements and enhancement adds to system capability by incorporating new requirements.*
- *Thus, maintenance changes the existing system, enhancement adds features to the existing system, and development replaces the existing system. It is an important part of system development that includes the activities which corrects errors in system design and implementation, updates the documents, and tests the data.*



# System Maintenance / Enhancement

## ❑ Maintenance Types

System maintenance can be classified into three types –

### ❖ Corrective Maintenance –

- *Enables user to carry out the repairing and correcting leftover problems.*
- *This is debugging the program, a process that continues throughout the life of the system.*

### ❖ Adaptive Maintenance –

- *Enables user to replace the functions of the programs.*
- *This is updating the system to accommodate changes in business conditions.*

### ❖ Perfective Maintenance –

- *Enables user to modify or enhance the programs according to the users' requirements and changing needs.*
- *This adds new functionality to the system.*
- *This process involves adding new features to the existing system without disturbing its operation*



# SDLC Methodology

- The life cycle defines a methodology for improving the quality of the system and the overall development process.
- A methodology is a formalized approach to implementing the SDLC (i.e., it is a list of steps and deliverables).
- There are many different systems development methodologies, and they vary in terms of the progression that is followed through the phases of the SDLC





# ISDL Methodologies

- Waterfall Model
- Iterative Model
- Spiral Model
- V-Model
- Big Bang Model

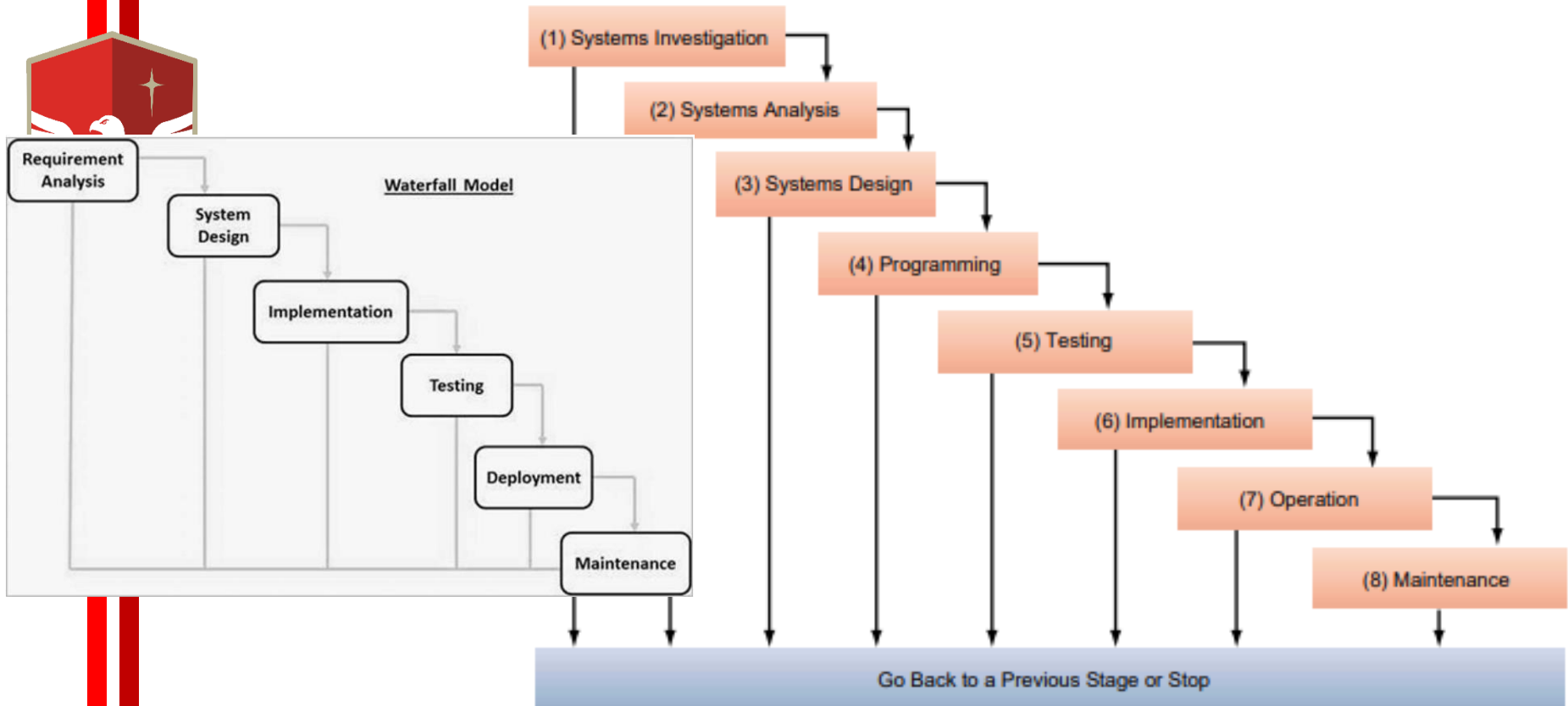
## *Other Related System Development Methodologies*

- *Architected Rapid Application Development (Architected RAD)*
- *Dynamic Systems Development Methodology (DSDM)*
- *Joint Application Development (JAD)*
- *Information Engineering (IE)*
- *Rapid Application Development (RAD)*
- *Rational Unified Process (RUP)*
- *Structured Analysis and Design*
- *eXtreme Programming (XP)*



# Waterfall Model

- The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**.
- Each phase must be completed before the next phase can begin and there is no overlapping in the phases.



# Waterfall Model – System Analysis

• **Systems Analysis** - is the examination of the business problem that the organization plans to solve with an information system.

This stage defines the business problem, identifies its causes, specifies the solution, and identifies the information requirements that the solution must satisfy.

Understanding the business problem requires understanding the various processes involved.

Organizations have three basic solutions to any business problem:

- (1) *Do nothing and continue to use the existing system unchanged.*
- (2) *Modify or enhance the existing system.*
- (3) *Develop a new system.*

The following information are produced:

- (1) *Strengths and weaknesses of the existing system*
- (2) *Functions that the new system must have to solve the business problem*
- (3) *User information requirements for the new system*



# Waterfall Model – Advantages

- The advantages of waterfall development are that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.
- Major advantages includes the following –
  - *Simple and easy to understand and use*
  - *Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.*
  - *Phases are processed and completed one at a time.*
  - *Works well for smaller projects where requirements are very well understood.*
  - *Clearly defined stages.*
  - *Well understood milestones.*
  - *Easy to arrange tasks.*
  - *Process and results are well documented.*



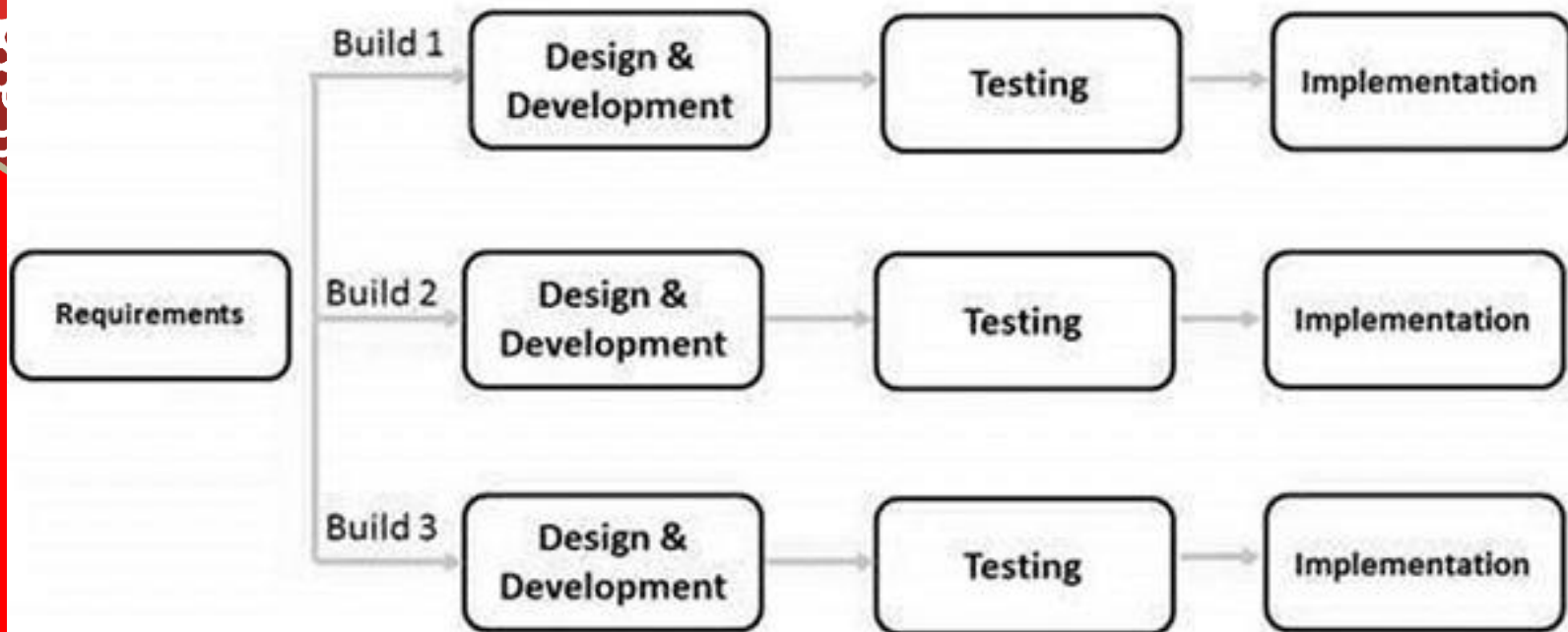
# Waterfall Model – Disadvantages

- The disadvantage of waterfall development is that it does not allow much reflection or revision.
- Major disadvantages are:
  - *No working system is produced until late during the life cycle.*
  - *High amounts of risk and uncertainty.*
  - *Not a good model for complex and object-oriented projects.*
  - *Poor model for long and ongoing projects.*
  - *Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model.*
  - *It is difficult to measure progress within stages.*
  - *Cannot accommodate changing requirements.*
  - *Adjusting scope during the life cycle can end a project.*
  - *Integration is done as a "big-bang. at the very end, which doesn't allow identifying any technological or business bottleneck or challenges early.*



# Iterative Model

- Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented.
- At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).



# Iterative Model – Advantages

- The advantage of this model is that there is a working model of the system at a very early stage of development, which makes it easier to find functional or design flaws.
  - *Some working functionality can be developed quickly and early in the life cycle.*
  - *Results are obtained early and periodically.*
  - *Parallel development can be planned and Progress can be measured.*
  - *Less costly to change the scope/requirements.*
  - *Testing and debugging during smaller iteration is easy.*
  - *Risks are identified and resolved during iteration; and each iteration is an easily managed milestone.*
  - *Easier to manage risk - High risk part is done first.*
  - *With every increment, operational product is delivered.*
  - *Risk analysis is better and It supports changing requirements.*
  - *Initial Operating time is less.*
  - *During the life cycle, software is produced early which facilitates customer evaluation and feedback.*



# Iterative Model – Disadvantages

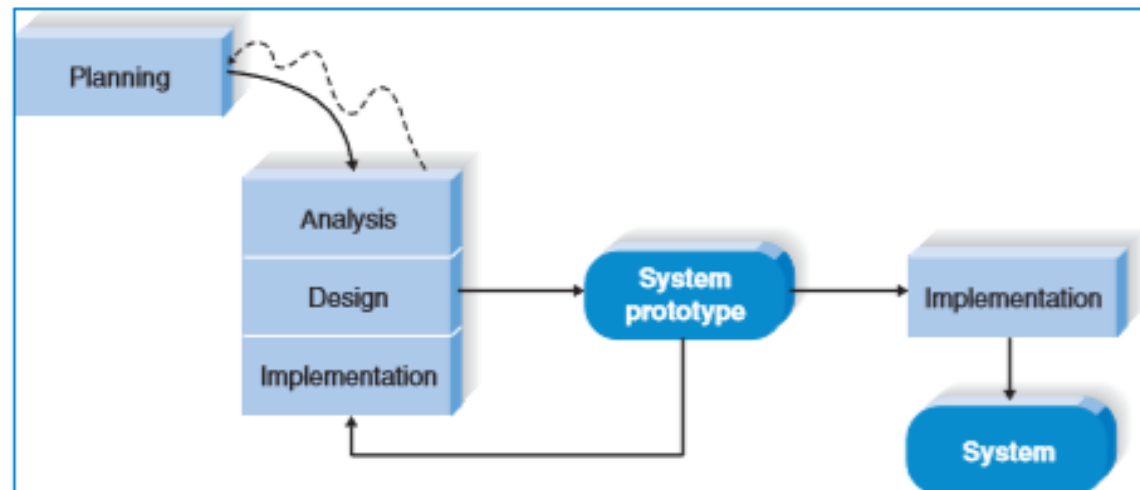
- The disadvantage with this SDLC model is that it is applicable only to large and bulky software development projects. This is because it is hard to break a small software system into further small serviceable increments/modules.
  - *More resources may be required.*
  - *Although cost of change is lesser, but it is not very suitable for changing requirements.*
  - *More management attention is required.*
  - *System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.*
  - *Defining increments may require definition of the complete system.*
  - *Not suitable for smaller projects.*
  - *Management complexity is more.*
  - *End of project may not be known which is a risk.*
  - *Highly skilled resources are required for risk analysis.*
  - *Projects progress is highly dependent upon the risk analysis phase.*





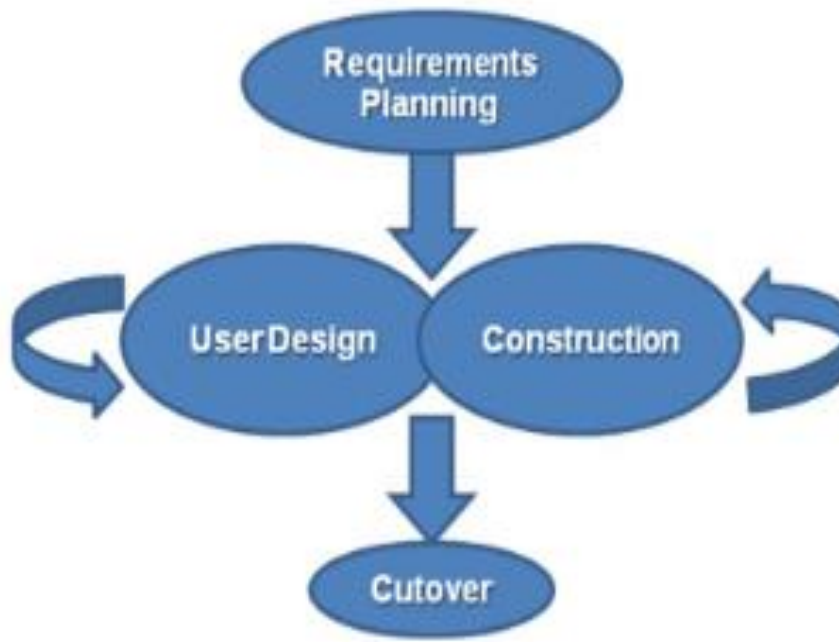
# Rapid Application Development Model

- The **RAD** model is based on prototyping and iterative development with no specific planning involved.
  - *The prototype approach provides a general idea of the user requirements. That is, the developers do not try to obtain a complete set of user specifications for the system and do not plan to develop the system all at once. Instead, they quickly develop a prototype, which either contains parts of the new system of most interest to the users, or is a small-scale working model of the entire system.*
  - *The prototype is given to the users, who are able to use it and make suggestions for improving it.*



# Rapid Application Development Model

- Rapid Application Development focuses on gathering customer requirements through workshops or focus groups, early testing of the prototypes by the customer using iterative concept, reuse of the existing prototypes (components), continuous integration and rapid delivery.
- RAD is a systems-development methodology that focuses on quickly building a working model of the system, getting feedback from users, and then using that feedback to update the working model.



Construction. the application developers, working with the users, build the next version of the system. This step is executed in parallel with the User Design step in an iterative fashion, until an acceptable version of the product is developed.

# RAD Model Design

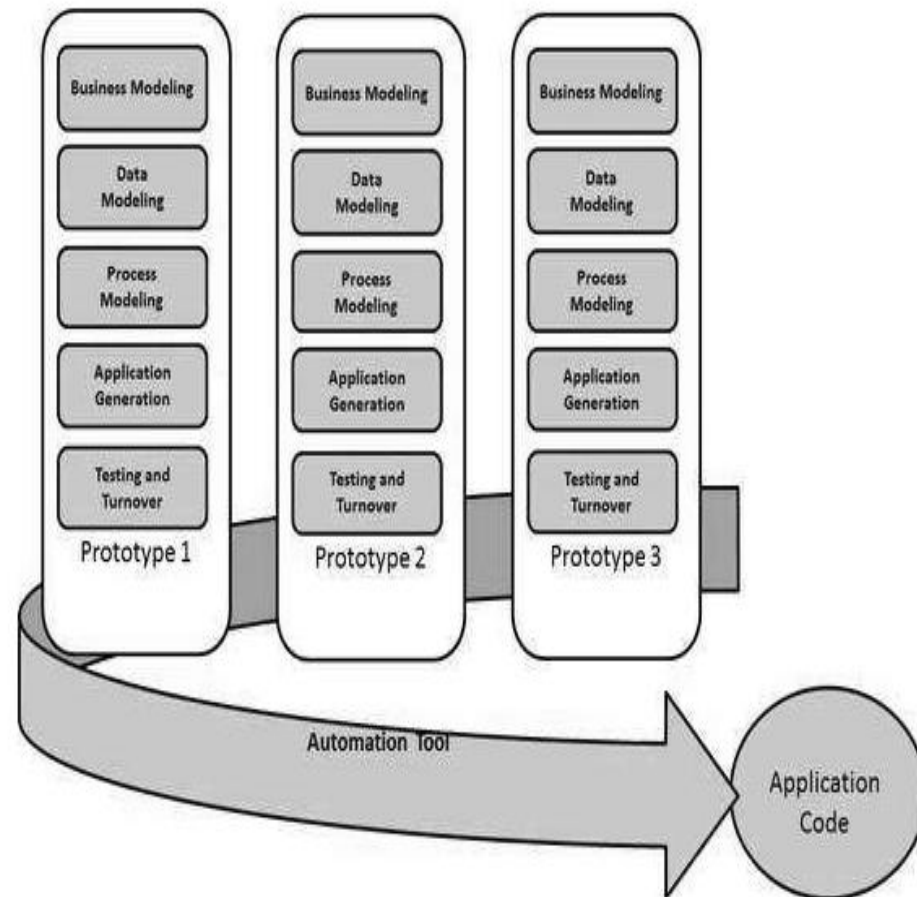
- Business Modeling - designed in terms of flow of information and the distribution of information between various business channels.
- Data Modeling - The information gathered in the BM phase is reviewed and analyzed to form sets of data objects vital for the business.

➤ Process Modeling - Process descriptions for adding, deleting, retrieving or modifying a data object are given.

➤ Application Generation- The actual system is built and coding is

➤ Testing and Turnover- However, the data flow and the interfaces between all the components need to be thoroughly tested with complete test coverage.

Since most of the programming components have already been tested, it reduces the risk of any major issues.



# RAD Model Pros and Cons

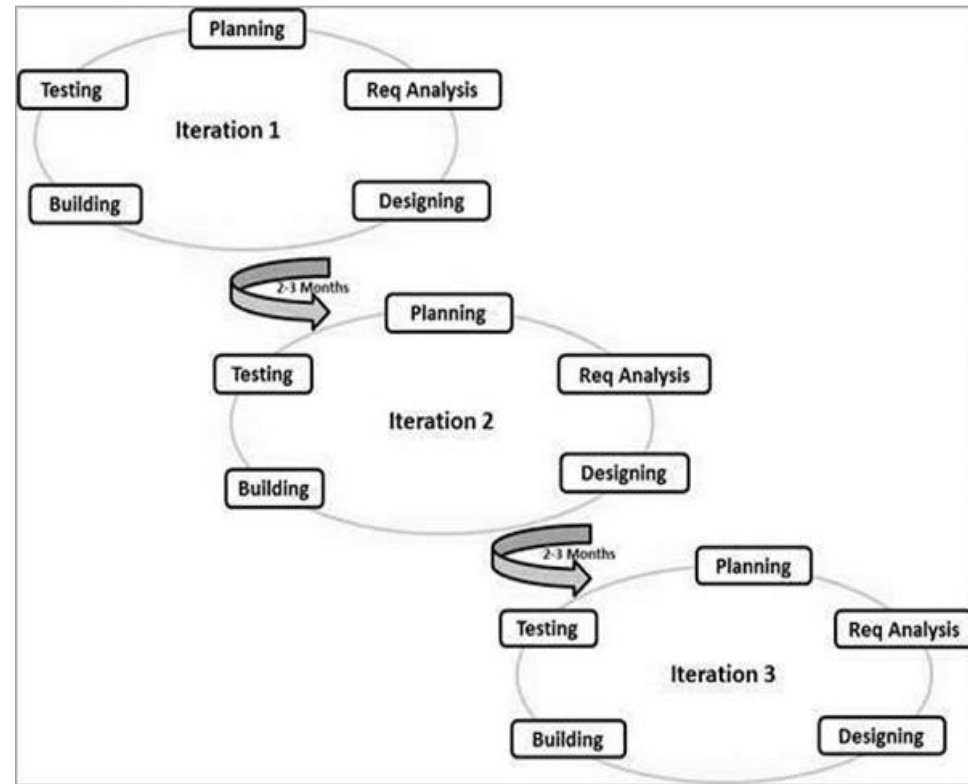
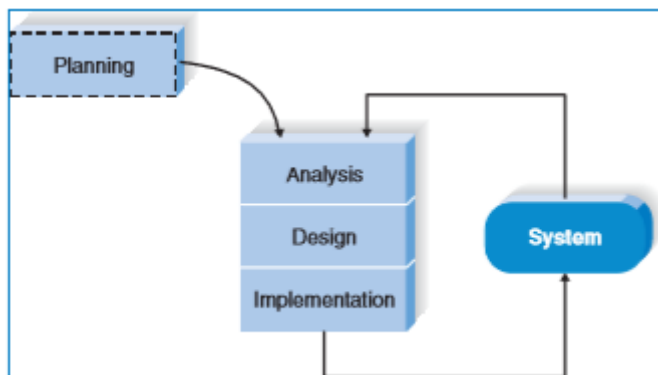
- Changing requirements can be accommodated.
- Progress can be measured.
- Productivity with fewer people in a short time.
- Reduced development time.
- Increases reusability of components.
- Quick initial reviews occur.
- Encourages customer feedback.
- Integration from very beginning solves a lot of integration issues
- Data Modeling - The information gathered in the BM phase is reviewed and analyzed to form sets of data objects vital for the business.
- Dependency on technically strong team members for identifying business requirements.
- Requires highly skilled developers / designers.
- High dependency on modeling skills.
- Inapplicable to cheaper projects as cost of modeling and automated code generation is very high.
- Management complexity is more.
- Suitable for systems that are component based and scalable.
- Requires user involvement throughout the life cycle.
- Suitable for project requiring shorter development times.



# AGILE Model

- Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.
- A group of programming-centric methodologies that focus on streamlining the SDLC.

Is a group of methodologies that utilize incremental changes with a focus on quality and attention to detail. Each increment is released in a specified period of time (called a time box), creating a regular release schedule with very specific objectives.



# Selecting the Appropriate Development Methodology

- Business Modeling - designed in terms of flow of information and the distribution of information between various business channels.
- Selecting the Appropriate Development Methodology

Usefulness In Developing Systems	Waterfall	Parallel	V-Model	Iterative	System Prototyping	Throwaway Prototyping	Agile Development
with unclear user requirements	Poor	Poor	Poor	Good	Excellent	Excellent	Excellent
with unfamiliar technology	Poor	Poor	Poor	Good	Poor	Excellent	Poor
that are complex	Good	Good	Good	Good	Poor	Excellent	Poor
that are reliable	Good	Good	Excellent	Good	Poor	Excellent	Good
with short time schedule	Poor	Good	Poor	Excellent	Excellent	Good	Excellent
with schedule visibility	Poor	Poor	Poor	Excellent	Excellent	Good	Good

# Appreciation

- Thank you all for your time



➤ God richly bless you

➤ Keep observing the COVID-19 protocols