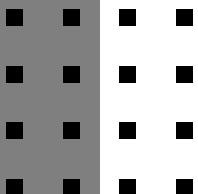


# Software testing

عرفان بهرامی - محمد مسین امتشامی - محمد سقلی - سیاوش قاسمی

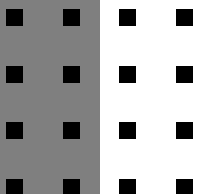
# Introduction

- It is the process used to identify the correctness, completeness and quality of developed computer software.
- It is the process of executing a program/application under positive and negative conditions by manual or automated means. It checks for the :
  - ❖ Specification
  - ❖ Functionality
  - ❖ Performance



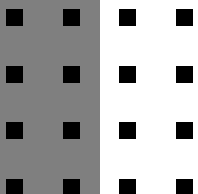
## Objectives

- Uncover as many as errors (or bugs) as possible in a given product.
- Demonstrate a given software product matching its requirement specifications.
- Validate the quality of a software testing using the minimum cost and efforts.
- Generate high quality test cases, perform effective tests, and issue correct and helpful problem reports.



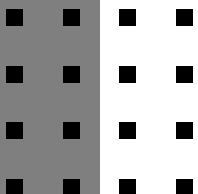
# Error, Bug, Fault & Failure

- **Error** : It is a human action that produces the incorrect result that produces a fault.
- **Bug** : The presence of error at the time of execution of the software.
- **Fault** : State of software caused by an error.
- **Failure** : Deviation of the software from its expected result. It is an event.

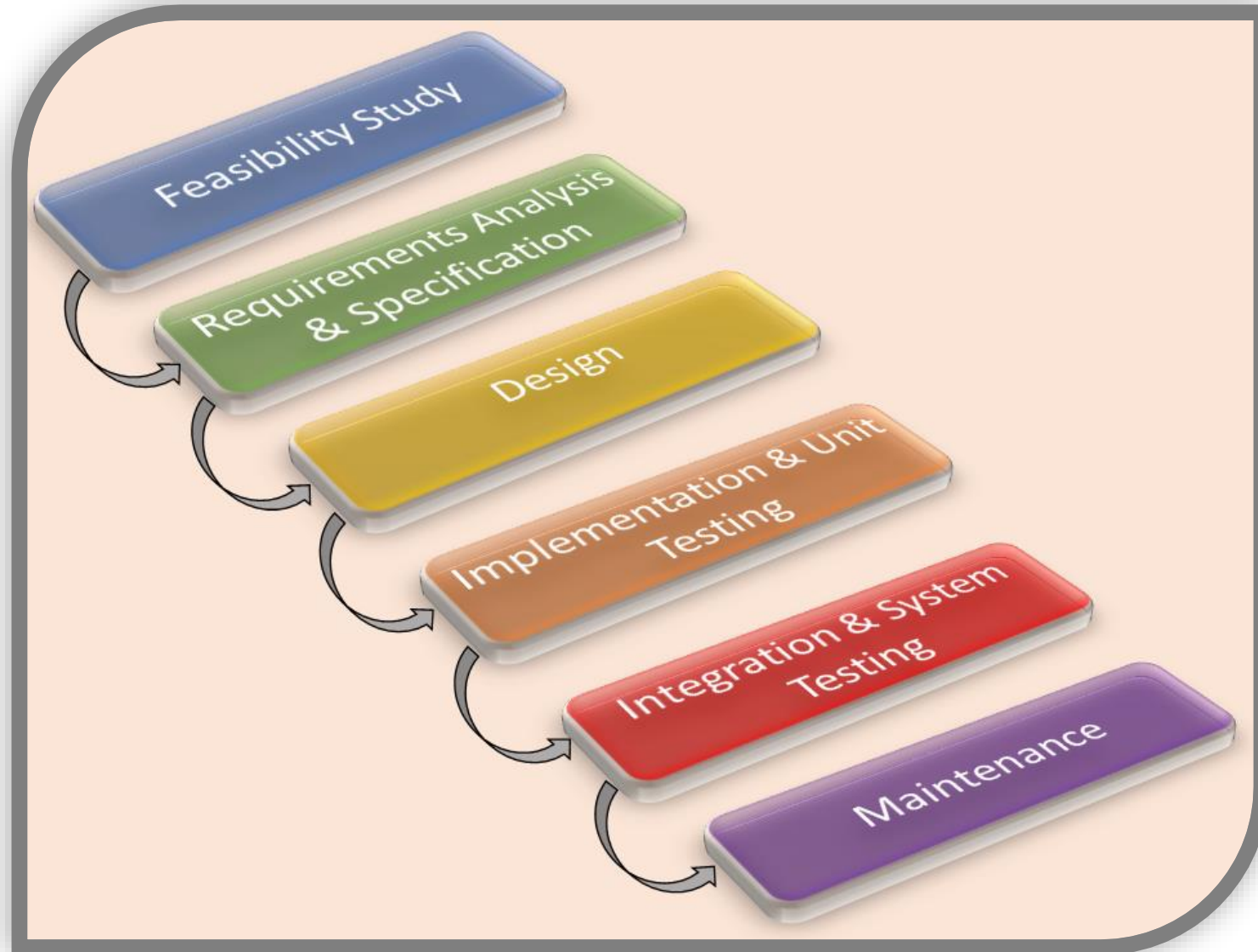


# SDLC(Software Development Life Cycle)

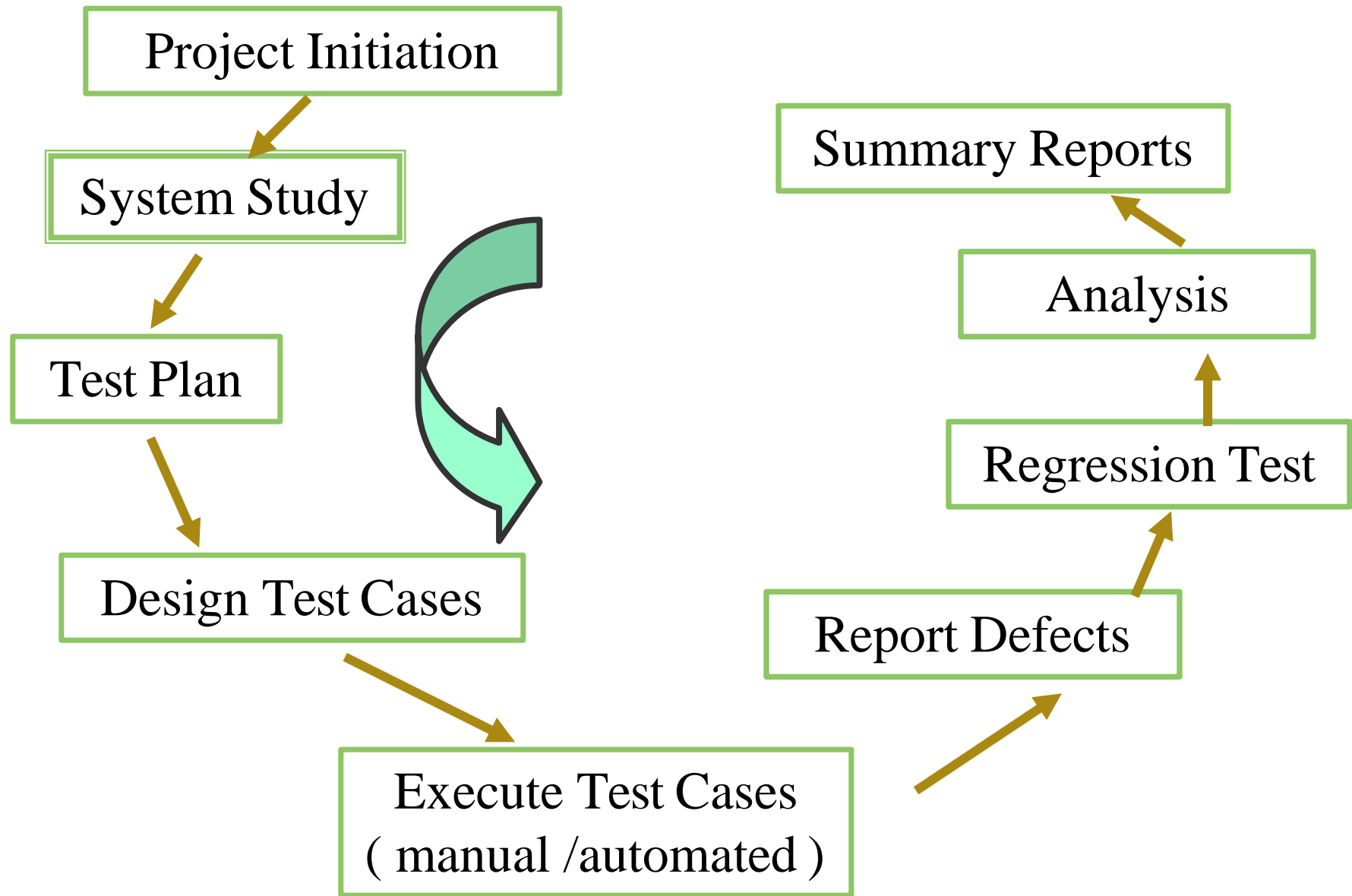
- Standard model used world wide to develop a software.
- A framework that describes the activities performed at each stage of a software development project.
- Necessary to ensure the quality of the software.
- Logical steps taken to develop a software product.



# Classical Waterfall Model



It is the oldest and most widely used model in the field of software development.



## Testing Life Cycle

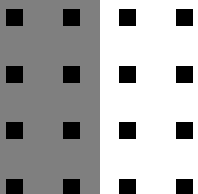
# Test Plan and Test Case

## Test Plan:

- It is a systematic approach to test a system i.e. software. The plan typically contains a detailed understanding of what the eventual testing workflow will be.

## Test Case:

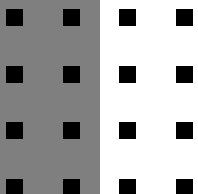
- It is a specific procedure of testing a particular requirement. It will include:
  - ❖ Identification of specific requirement tested
  - ❖ Test case success/failure criteria
  - ❖ Specific steps to execute test
  - ❖ Test data





# Verification vs Validation

- **Verification:** The software should confirm to its specification  
(Are we building the product right?)
- **Validation:** The software should do what the user really requires  
(Are we building the right product?)



# Testing Methodologies

## ➤ **Black box testing :**

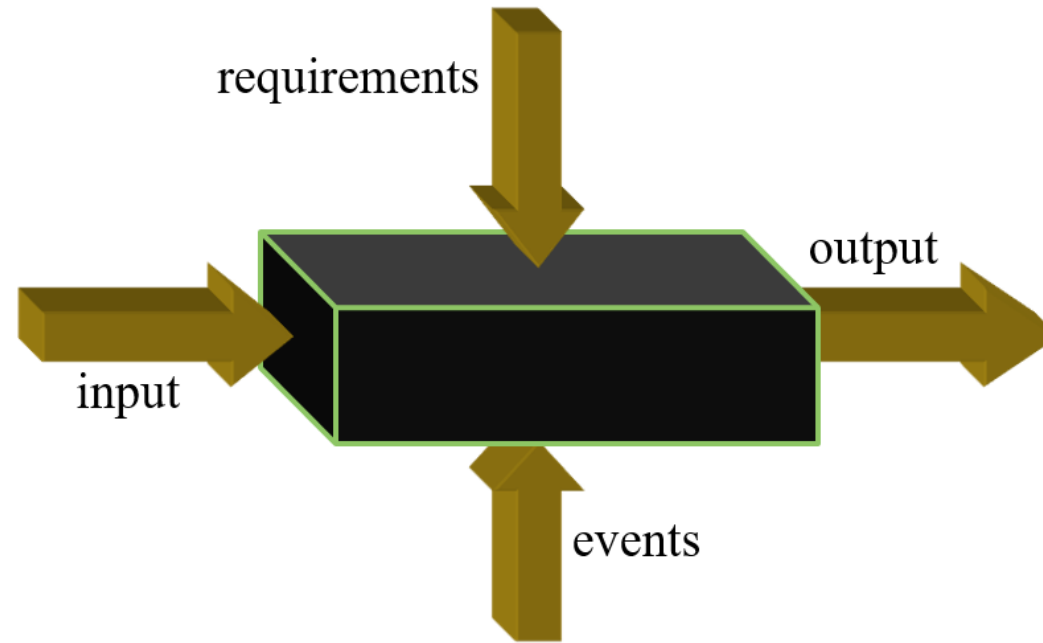
- No knowledge of internal program design or code required.
- Tests are based on requirements and functionality.

## ➤ **white box testing :**

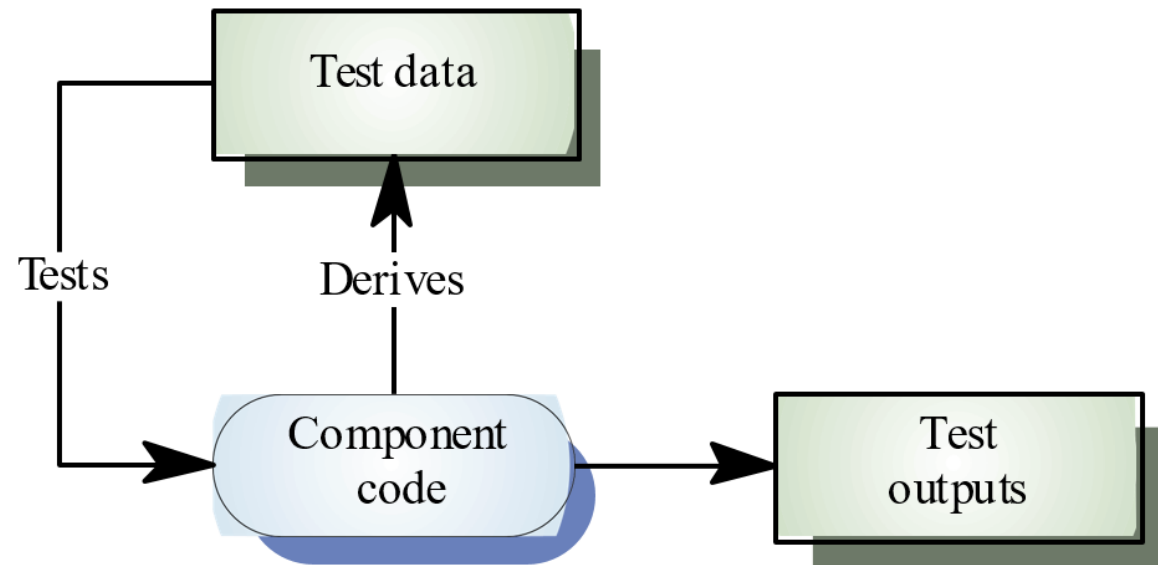
- Knowledge of the internal program design and code required.
- Tests are based on coverage of code statements, branches, paths, conditions.



black box



white box



# Testing Levels

## *Levels of Testing*

Unit Test

Test Individual Component

Integration  
Test

Test Integrated Component

System Test

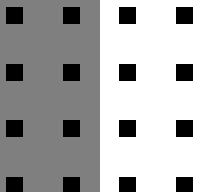
Test the entire System

Acceptance  
Test

Test the final System

# UNIT TESTING

- ☐ Tests each module individually.
- ☐ Follows a white box testing (Logic of the program).
- ☐ Done by developers.



# INTEGRATION TESTING

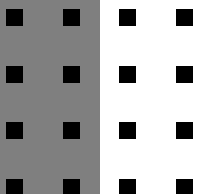
- ☐ Once all the modules have been unit tested, integration testing is performed.
- ☐ It is systematic testing.
- ☐ Produce tests to identify errors associated with interfacing.

## **Types:**

- Big Bang Integration testing
- Top Down Integration testing
- Bottom Up Integration testing
- Mixed Integration testing

# SYSTEM TESTING

- ❑ System testing is the first level in which **the complete application is tested as a whole.**
- ❑ The goal at this level is to evaluate whether the system has complied with all of the outlined requirements and to see that it meets Quality Standards.
- ❑ System testing is undertaken by independent testers who haven't played a role in developing the program.
- ❑ This testing is performed in an environment that closely mirrors production.



# ACCEPTANCE TESTING

- ❑ **ACCEPTANCE TESTING** is a level of software testing where a system is tested for acceptability.
- ❑ The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.
- ❑ Done by end users

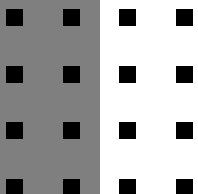
## Types:

- User acceptance testing
- Operational acceptance testing
- Contractual and regulatory acceptance testing
- Alpha and beta testing



## DISCUSSION

- In order to be cost effective, the testing must be concentrated on areas where it will be most effective.
- The testing should be planned such that when testing is stopped for whatever reason, the most effective testing in the time allotted has already been done.
- The absence of an organizational testing policy may result in too much effort and money will be spent on testing, attempting to achieve a level of quality that is impossible or unnecessary.





THANK YOU