Coding and Logic

Describe the functions of each stage of the application lifecycle management (ALM)

Application Lifecycle Management

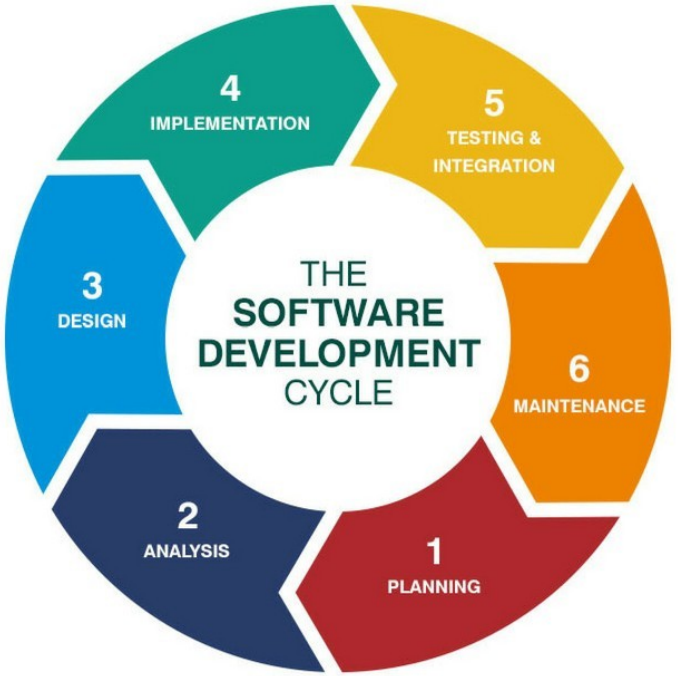
- application development phases
  - Requirements
  - Design
  - build
- service management phases
  - Deploy
  - Operate
  - optimise
- application management

ALM Definition

- Application Lifecycle Management covers the entire history of an application, utility, component or software solution, from the initial idea to its removal from an organization’s systems

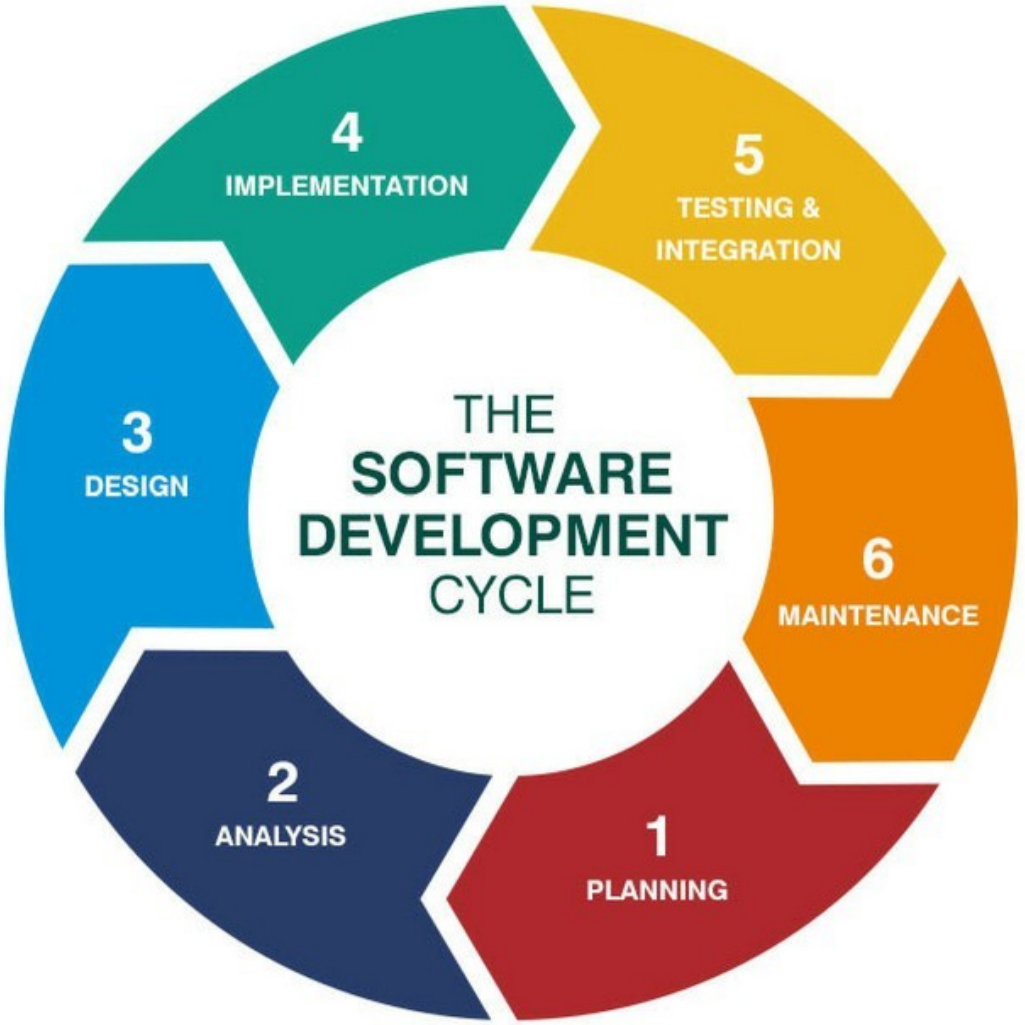
Software Development Life Cycle

- Part of ALM – Similar to but not the whole thing



Synotive


PLAN
DESIGN
DEVELOP
TEST
DEPLOY
MAINTAIN
SOFTWARE
DEVELOPMENT
LIFECYCLE


4
IMPLEMENTATION
5
TESTING &
INTEGRATION
3
DESIGN
THE
SOFTWARE
DEVELOPMENT
CYCLE
6
MAINTENANCE
2
ANALYSIS
1
PLANNING

Requirements

- Requirements – Definition of the system requirements to allow the developer(s) to understand what is required of the software system.
- Establish customer expectations and project scope
- In larger system all development teams must communicate to develop system interfaces.
  - Collect information and inventories of existing services.
  - Establish the business case for implementing the service

Analysis

- The requirements are analysed for financial viability, operational capability and technical feasibility of the system

Design

- Design:
- detailed definition of inputs and outputs
- Detailed definition of processing
- Uses tools such as UML (Unified Modelling Language).

Coding

- Writing the software to implement the design
  - quality assurance
  - Inspections
  - unit testing
  - integration testing.

Testing

- Once Coding is complete, use system tests to discover as many software errors as possible.
- Before the software is passed to the client.
- Client can carry out Customer Acceptance Testing (CAT).

Service Management

- Deployment
  - The software system is installed at the customers target site
  - If part of a larger system, it may be an upgrade
- Operation
  - Starts as soon as the software is installed
  - Monitor how successful the implementation is
  - Close contact with customer
  - Optimise for the customer

Maintenance

- Required over the life of the software
- Repair - fix a fault identified by the client
- Adapted - use the current features to fulfil new requirements
- Perfective - add new features to improve system performance and/or functionality