



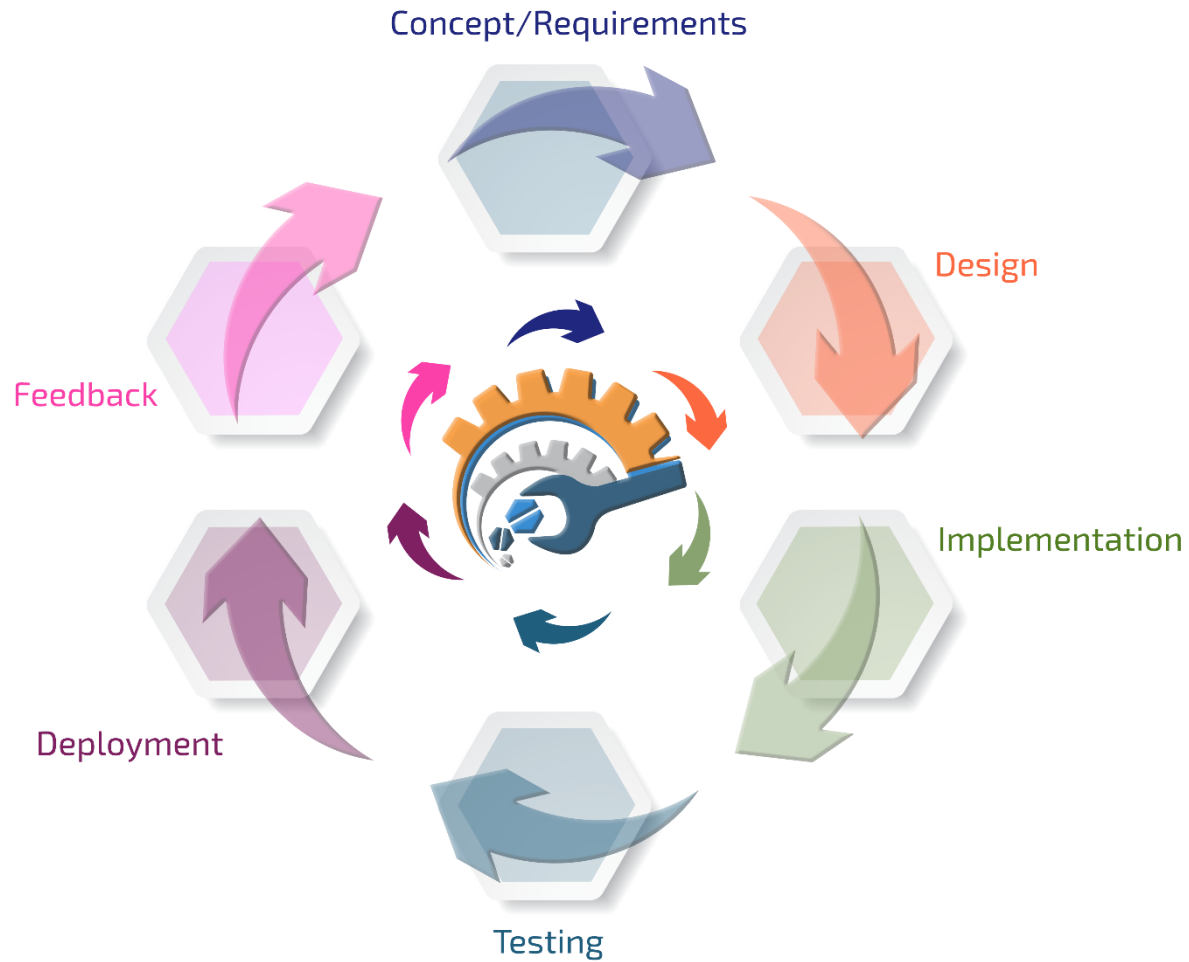
## **DewProcess**

Software Development  
Life Cycle

## DewProcess

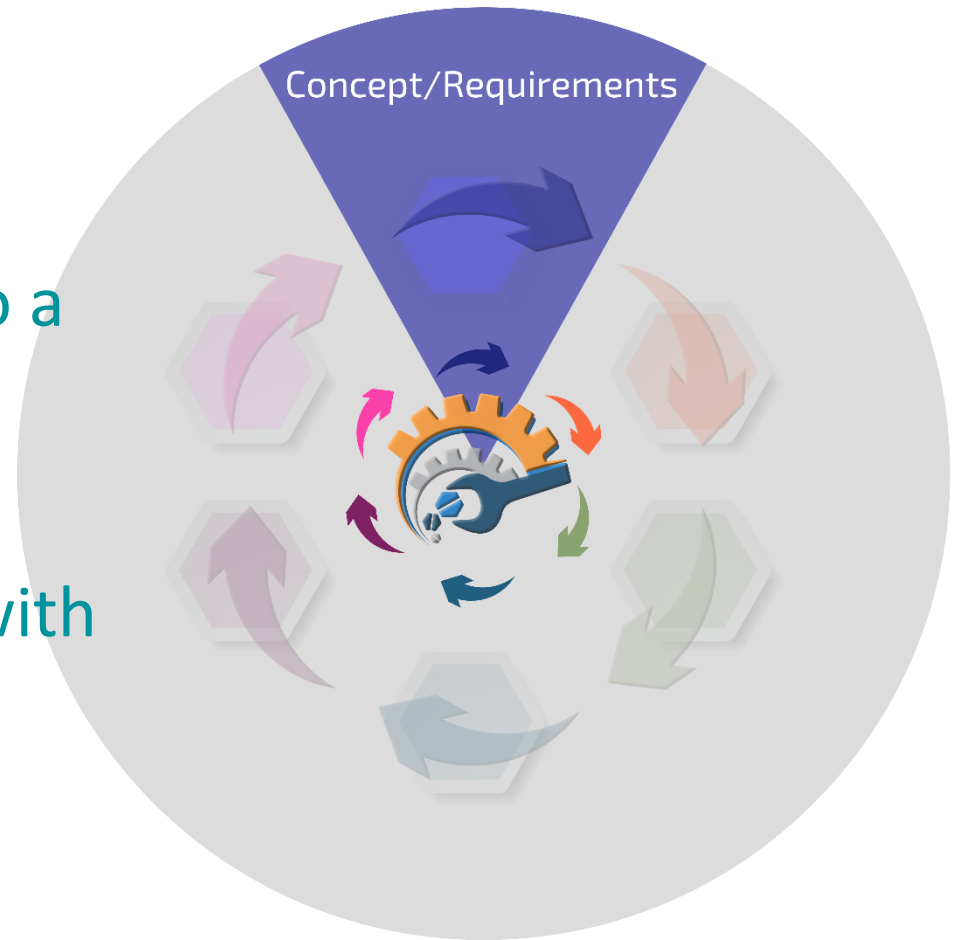
An adaptive process of delivering software by utilizing application delivery best practices as well as your organization's strengths to deliver best in class solutions.

# DewProcess



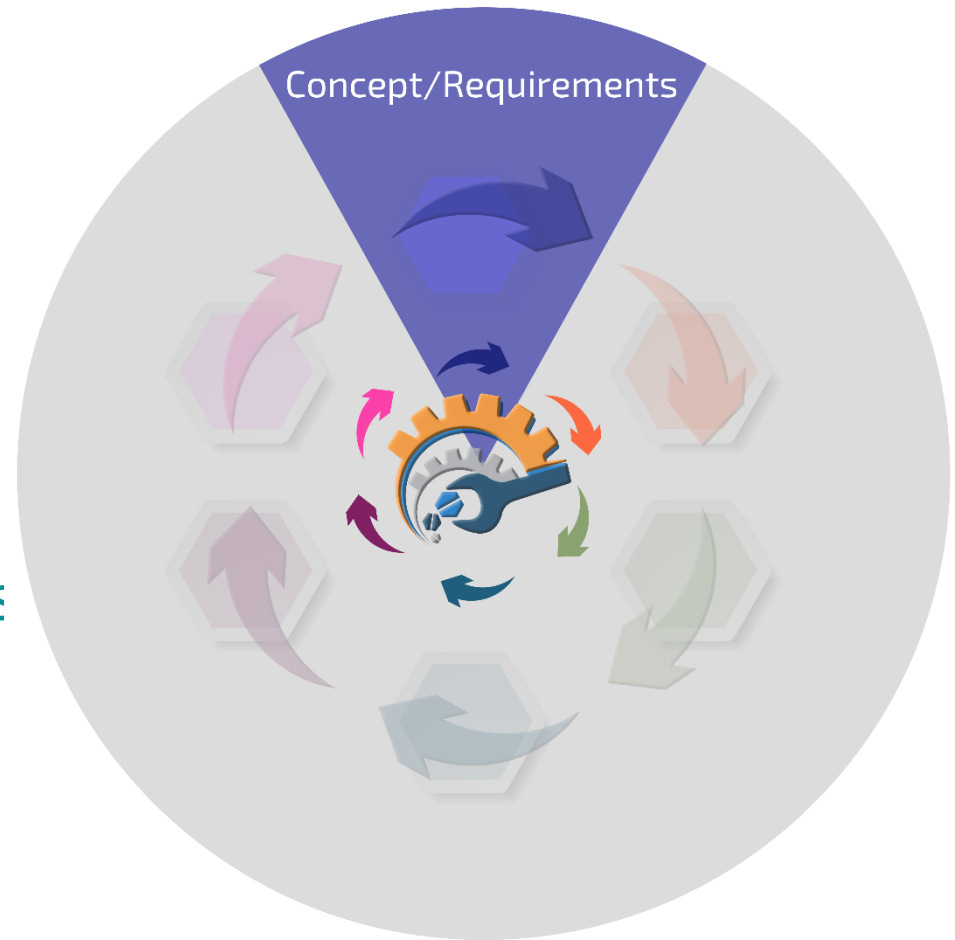
## Objective

- Understand business needs and translate into a solution
- Merge business vision with the proper technology solutions



## Deliverables

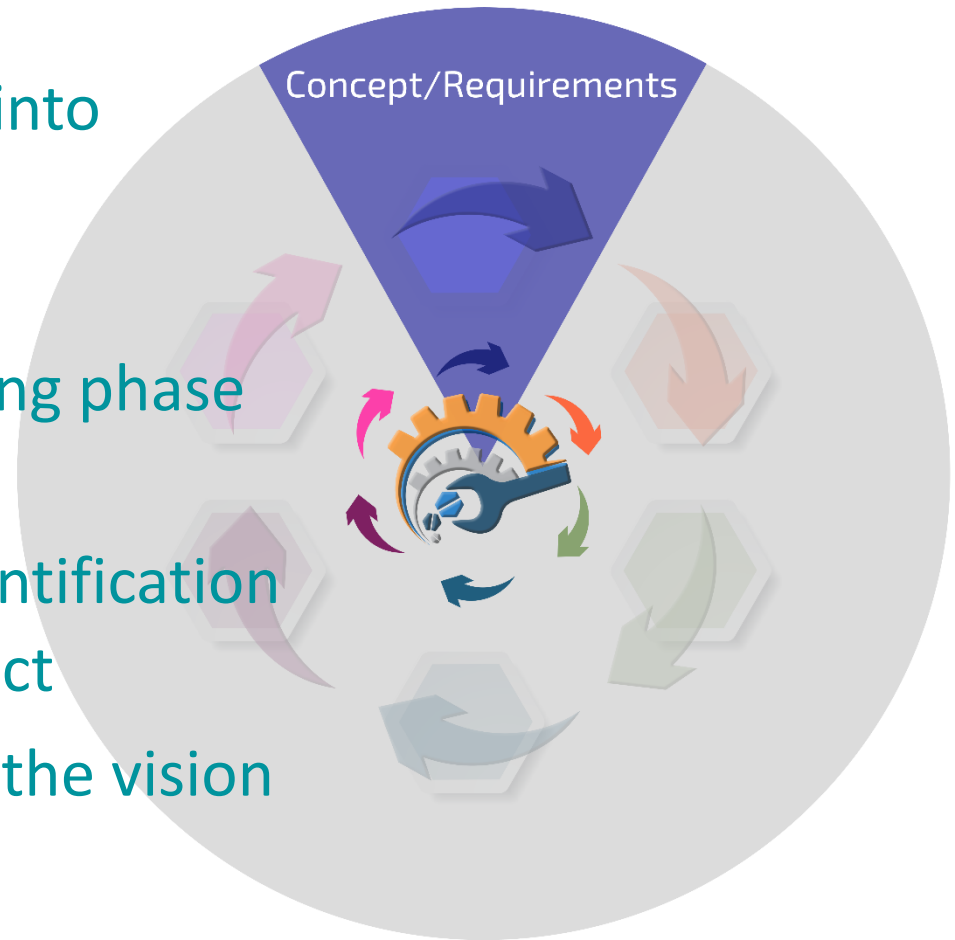
- High-level Problem Statement
- Strengths Weaknesses Opportunities Threats (SWOT) analysis
- Issues Risks Assumptions Decisions (IRAD) documentation
- Journey Map



# Requirements

## Objective

- Translate business needs into technical and functional requirements.
- The requirements gathering phase asks: **WHAT?**
- Emphasis is placed on identification of Minimum Viable Product
- Quickly begin delivery on the vision



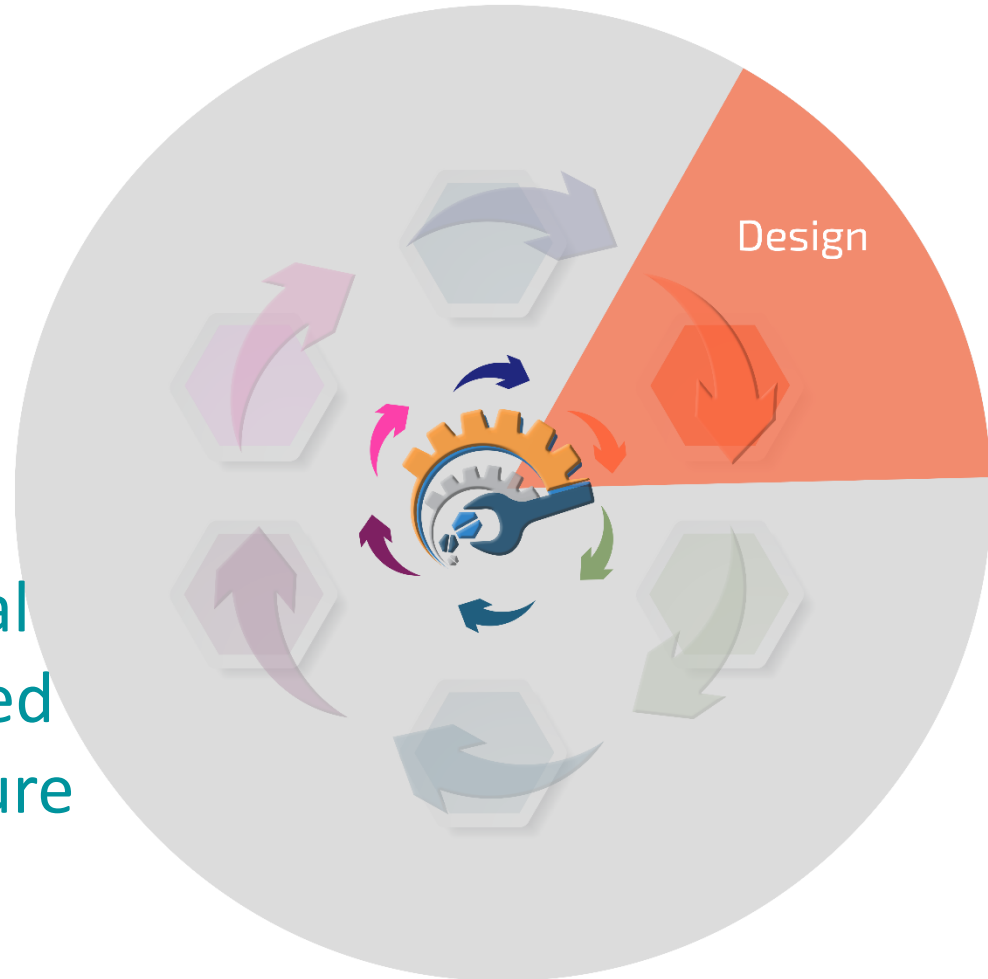
# Requirements

## Deliverables

- High-level details of the business capabilities for the project (Epics/BRDs)
- Groupings of business functions that make up each capability (Features/FRDs)
- Groupings of technical functions that developers can code to create the business functions (User Stories/TRDs)
  - Requirements meet the definition of ready Follow INVEST (Independent, Negotiable, Valuable, Estimate, Small, Testable)
  - Acceptance criteria defined - What is it supposed to do?
- Objective measurements that illustrate the completion of a requirement (acceptance criteria)

## Objective

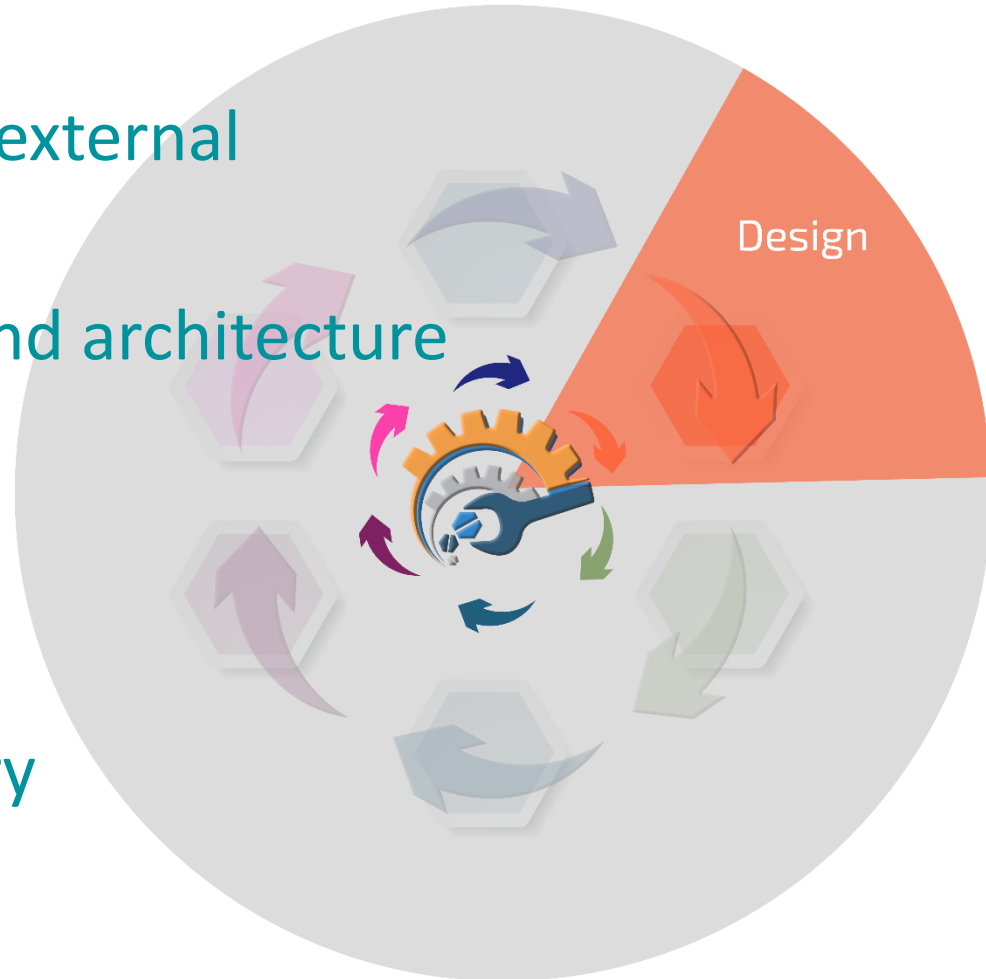
- Derives technical approach from concept and business need
- The design phase asks: **HOW?** The architectural foundation is determined and put in place to ensure rapid delivery.





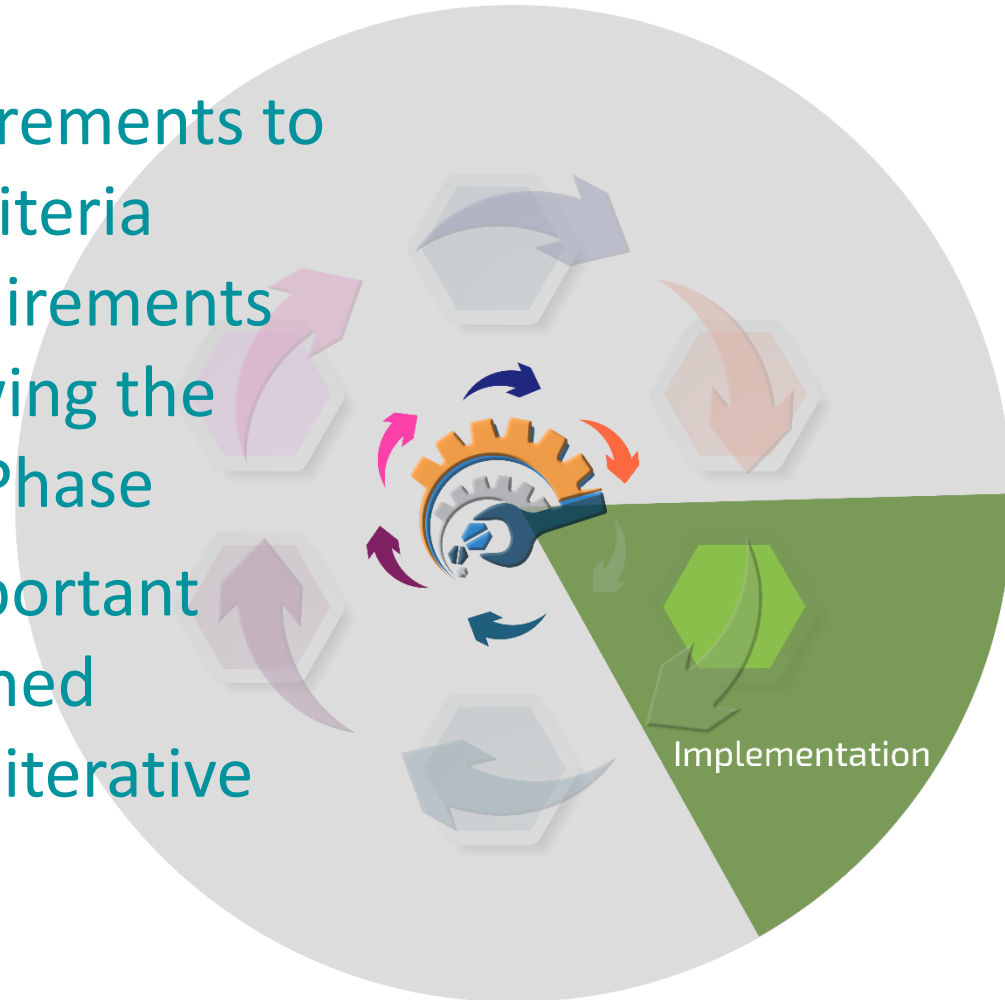
## Deliverables

- Integration points with external systems
- System infrastructure and architecture
- Data modeling
- Wireframes
- Design comps
- Requirements repository
- Release plan



## Objective

- Fulfilling business requirements to meet the acceptance criteria determined in the Requirements Gathering phase, following the patterns of the Design Phase
- Delivering the most important business value determined collaboratively through iterative development cycles



## Deliverables

- Working software
- Business value as defined in requirements
- Continuous feedback from stakeholders



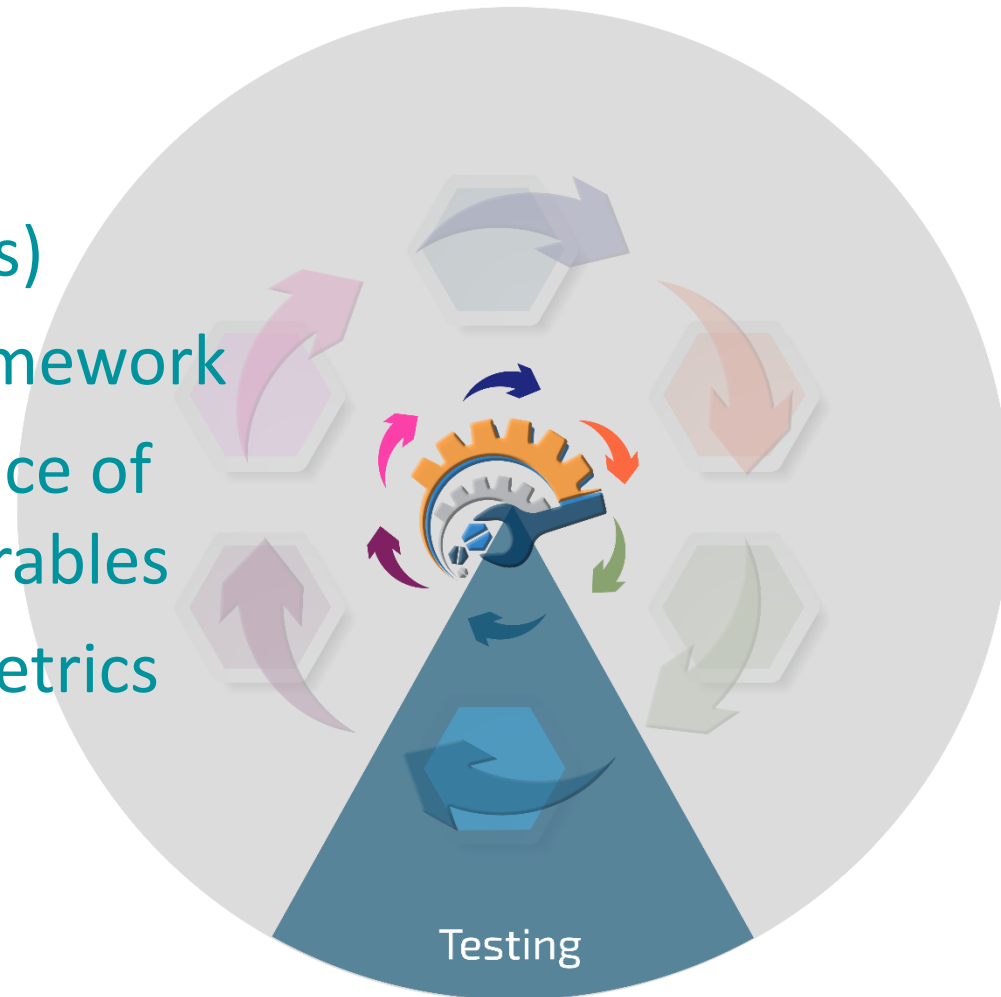
## Objective

To assert that the business value delivered in the Implementation phase meets the functional, technical and business requirements and meets the quality standards of the organization



## Deliverables

- Test results
- Test artifacts (test cases)
- Customized testing framework
- Business user acceptance of implementation deliverables
- Performance testing metrics
- Documentation



## Objective

Documentation and  
delivery of product to end-  
user



## Deliverables

- Release notes
- Working product
- Audit traceability



## Objective

Continually refine and prioritize business needs by providing clarity for highest priority features and capabilities being delivered in upcoming releases.





## Deliverables

- On-Going Collaboration Meetings
  - Weekly meetings to refine requirements with cross-functional team of
    - Product Owners
    - Application Delivery Leads
    - Business Analyst
    - Architects
- Requirements that are Ready for Application Delivery
  - Follow INVEST (Independent, Negotiable, Valuable, Estimate, Small, Testable)
  - Clear Acceptance Criteria
  - Clear understanding of the desired functionality

# Incremental Release Plans

## Objective

Determine what features and capabilities will be delivered in what order determined by priority, architectural significance and business value derived.

# Backlog Maintenance

## Product Backlog

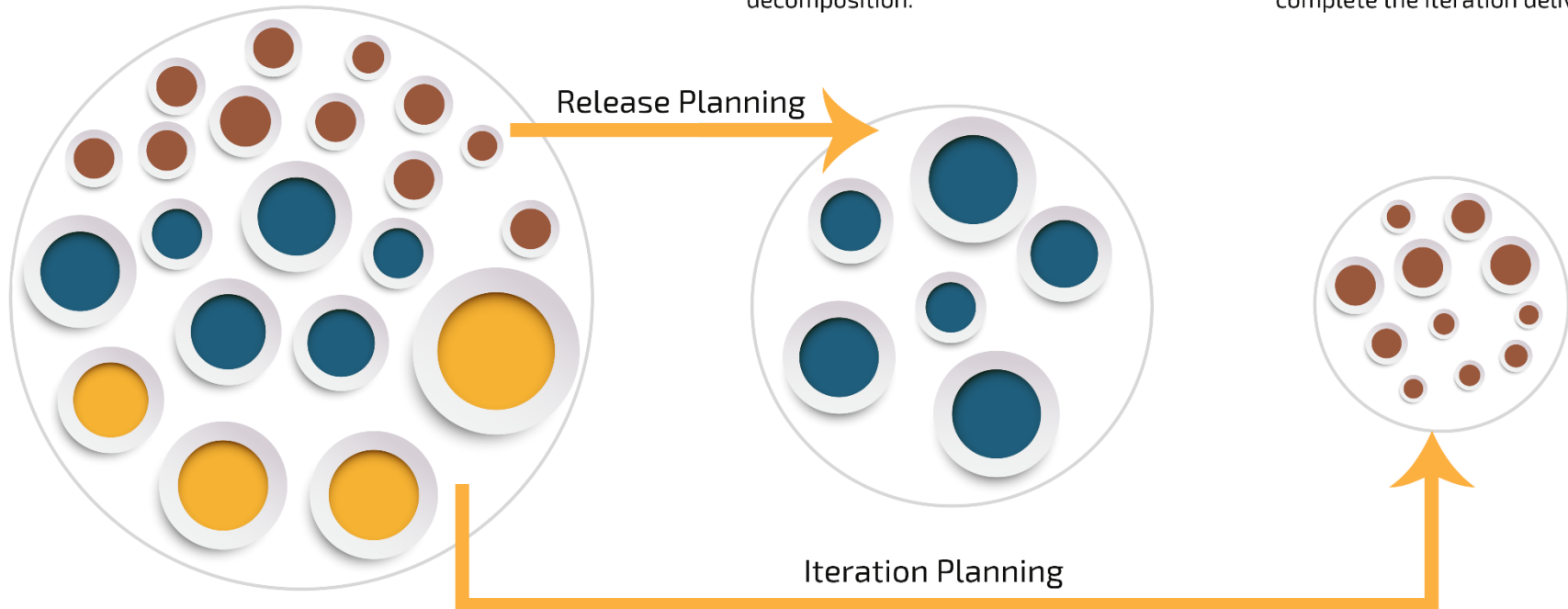
- The collection of features and functions required to build the product
- Living body of work that is used as the requirements for the product

## Release Backlog

- The collection of features and functions required to satisfy the terms of the next release
- Contains user stories and technical documents that likely require further decomposition.

## Iteration Backlog

- The collection of features and functions required to complete the agreed-upon work for the next 2-3 weeks.
- Well-groomed user stories with tasks defined to drive the work required to complete the iteration delivery.



# Incremental Release Plans

## Deliverables

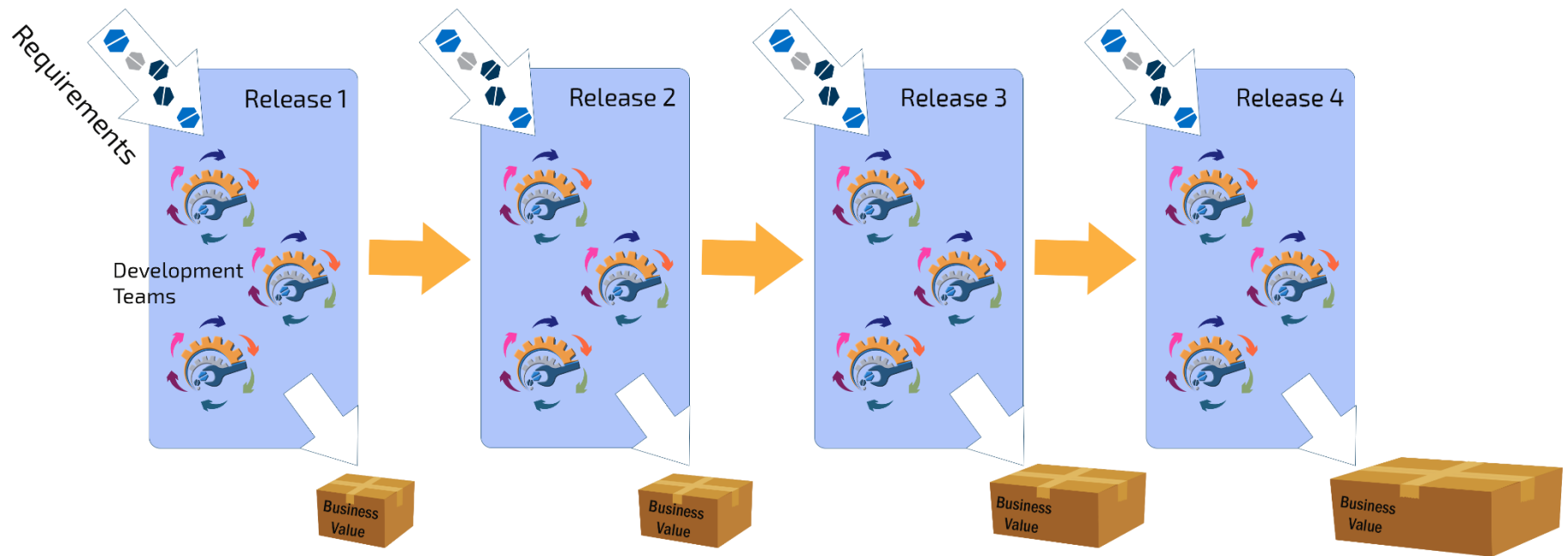
- Release Plan
  - Targeted features and capabilities
  - T-shirt sizing estimates
  - Release date(s)
  - Number of iterations
- Break Down Features into Workable Requirements
  - Follow INVEST (Independent, Negotiable, Valuable, Estimate, Small, Testable)
  - Estimated in Relative Complexity
- Sprint Plan
  - Number of iterations
  - Requirement per sprint
- Dependency Mapping

# Short Iterations with feedback loops

## Objectives

Deliver business value by fulfilling requirements and demonstrating them to elicit feedback and acceptance

# Short Iterations with feedback loops



# Short Iterations with Feedback Loops

## Deliverables

- Sprint Plan
  - Requirements commitment
  - Requirements estimated in relative complexity
  - Task defined with hour estimates
- Working Software - Showcase of Work Product
  - Demonstrate fulfillment of the requirements
  - Get acceptance of requirements or elicit improvements to deliverables
- Retrospectives
  - Document what improvements can be made to process and product

## Objective

Working software that meets the strategic need of the business problem statement