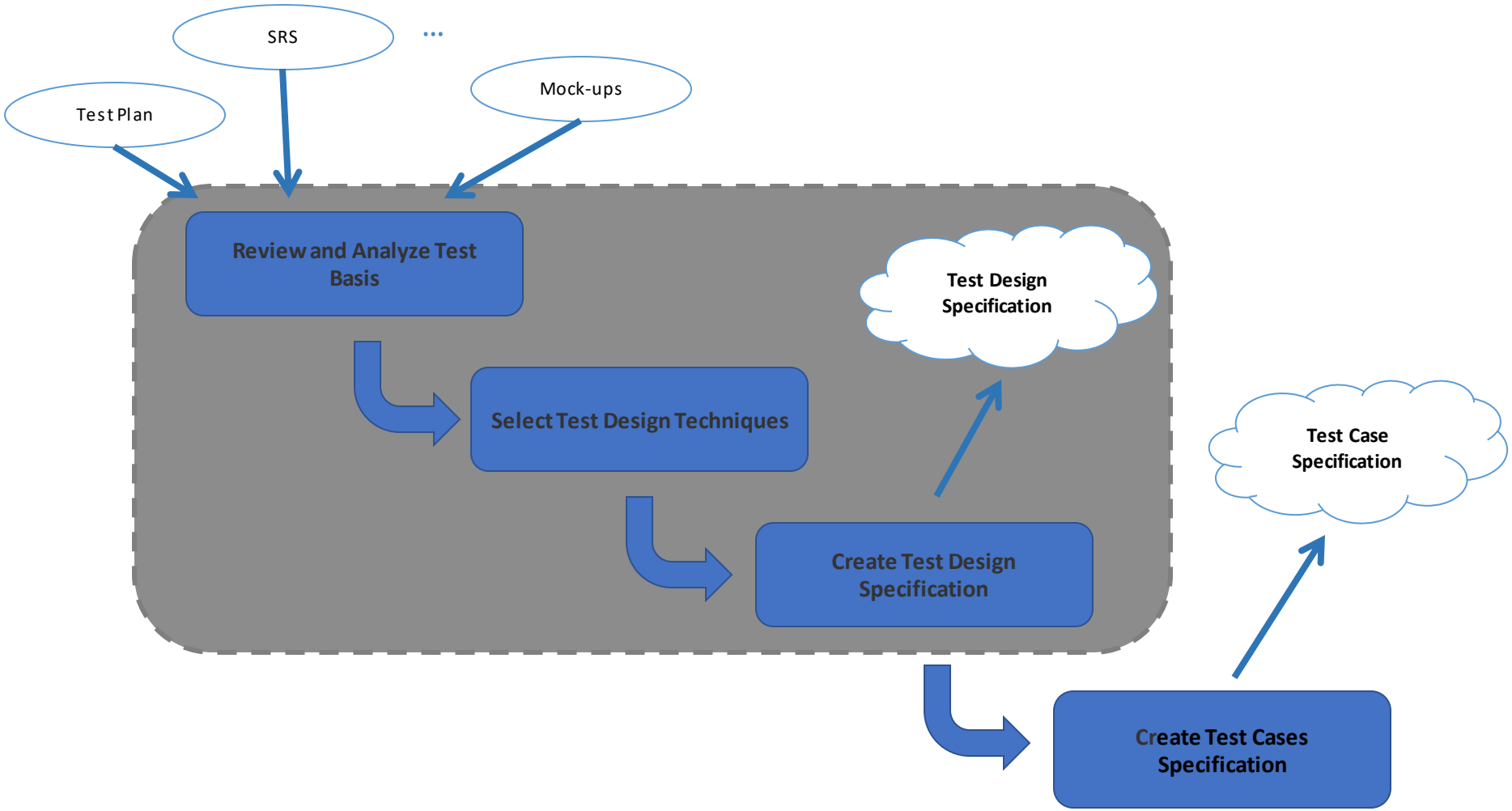
Test Design

- **Test Design Phase** – In software engineering, test design phase is a process of reviewing and analyzing test basis, selecting test design techniques and creating designed test cases, checklists and scenarios for testing software.

- **Test Design Specification**
  - It is a document that describes features to be tested and specifies list of all test scenarios or test cases, which should be designed for providing the testing of software.
  - The test design does not record the values to be entered for a test, but describes the requirements for defining those values.

- Test design could require all or one of:
  - Knowledge of the software, and the business area it operates on
  - Knowledge of the functionality being tested
  - Knowledge of testing techniques and heuristics
  - Planning skills to schedule in which order the test cases should be designed, given the effort, time and cost needed or the consequences for the most important and/or risky features

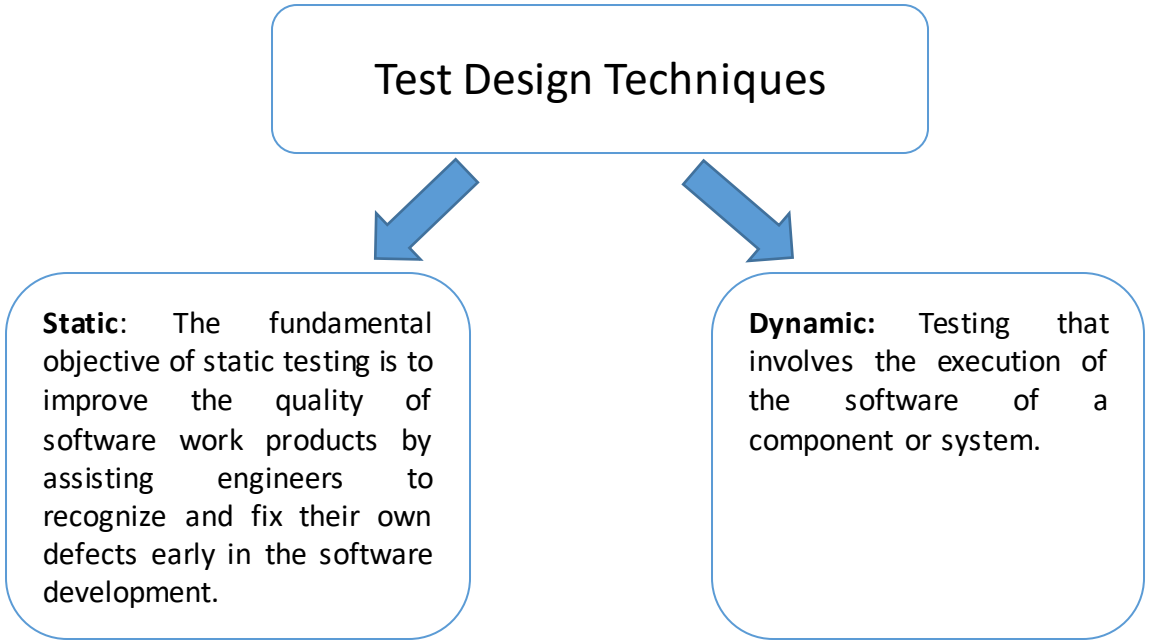

SRS
...
Test Plan
Mock-ups
Review and Analyze Test Basis
Select Test Design Techniques
Create Test Design Specification
Test Design Specification
Test Case Specification
Create Test Cases Specification

Test Analysis

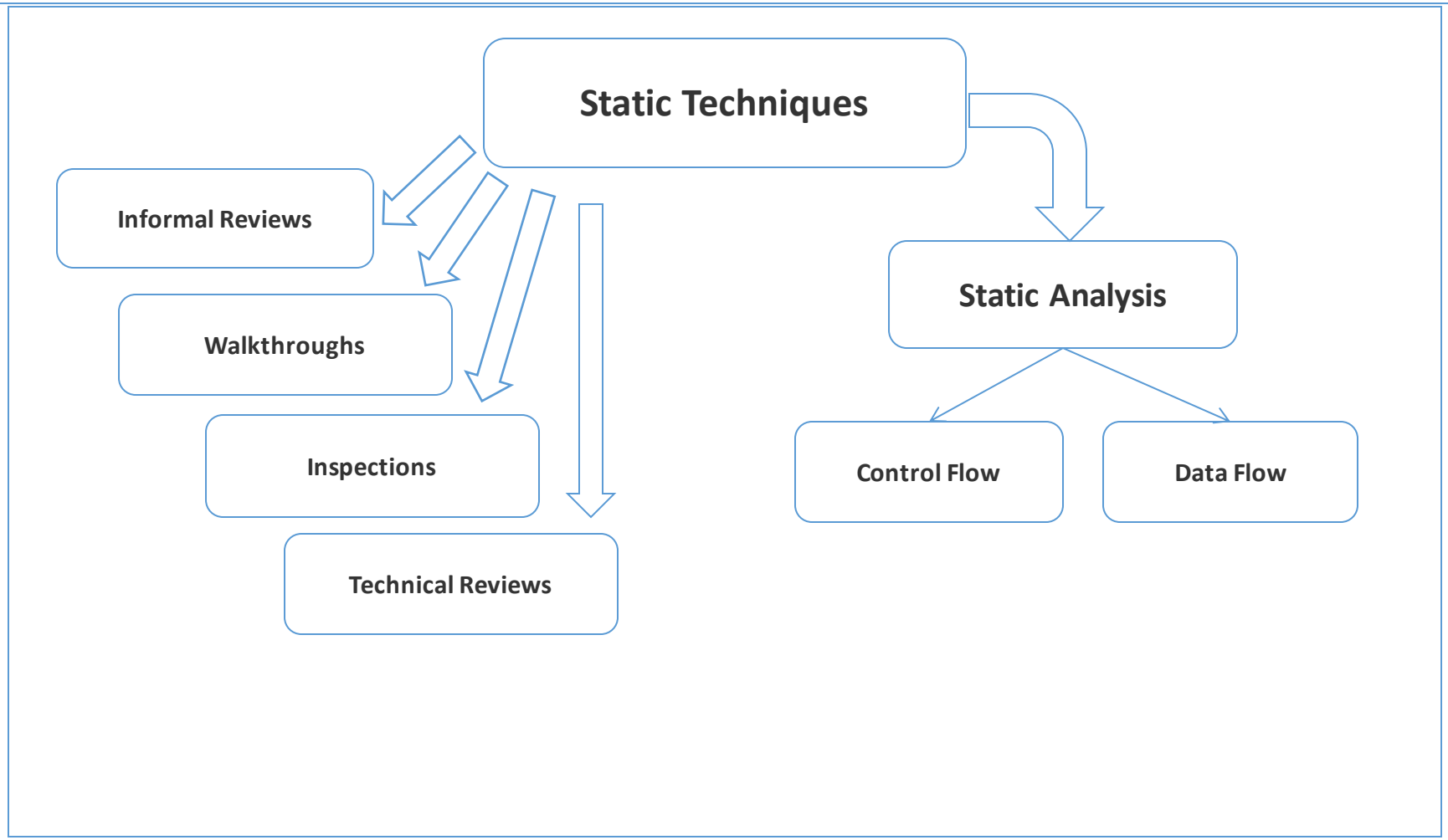
- **Test Analysis** is the process of looking at something that can be used to derive test information. This basis for the tests is called the 'test basis'.
- Test analysis has the following major tasks, in approximately the following order:
  - Review **Test Basis**
  - Define **Test Conditions**
  - Evaluate testability of the requirements and system
  - Define test environment
- **Test Basis** – all documents from which the requirements of a component or system can be inferred (the documentation on which the test cases are based).
- **Test Condition** – an item or event of a component or system that could be verified by one or more test cases, e.g. a function, transaction, feature, quality attribute or structural element.
- **Traceability** – the ability to identify related items in documentation and software, such as requirements with associated tests. There are:
  - Horizontal traceability
  - Vertical traceability

Test Design Techniques

- **Test Design Techniques** are used to derive and/or select test cases
- Why they are important?
- Two main categories of Test Design Techniques



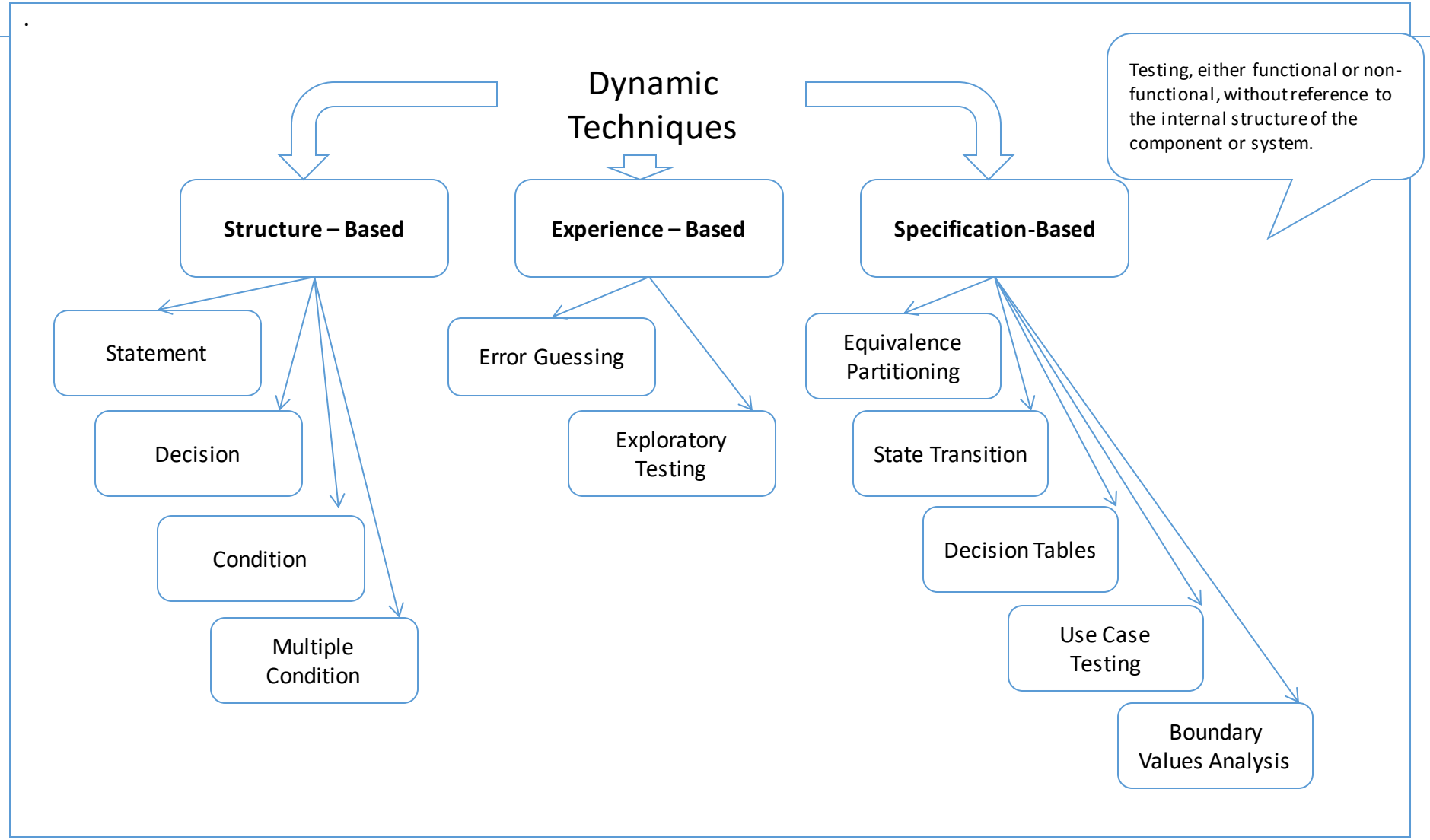
Static Techniques



Statique Techniques

- **Informal Review** – a review not based on a formal (documented) procedure.
- **Walkthrough** – a step-by-step presentation by the author of a document in order to gather information and to establish a common understanding of its content.
- **Technical Review** – a peer group discussion activity that focuses on achieving consensus on the technical approach to be taken.
- **Inspection** – a type of peer review that relies on visual examination of documents to detect defects. The most formal review technique and therefore always based on a documented procedure.
- **Control flow analysis** – a form of static analysis based on a representation of unique paths (sequences of events) in the execution through a component or system. Control flow analysis evaluates the integrity of control flow structures, looking for possible control flow anomalies such as closed loops or logically unreachable process steps.
- **Data Flow Analysis** – a form of static analysis based on the definition and usage of variables.

Dynamic Techniques



Test Design Specification Structure

- According to IEEE-829 standard template structure looks in the following way:

1. Test Design Specification Identifier
    - 1.1 Purpose
    - 1.2 References
    - 1.3 Definitions, acronyms and abbreviations
2. Features to be Tested
3. Approach Refinements
4. Test Identification
    - 4.1 <Test Item 1>
    - 4.2 <Test Item …>
    - 4.3 <Test Item N>
5. Feature Pass/Fail Criteria

Test Design Specification Structure

- **Test Design Specification Identifier** section covers:
  - Purpose of the document
  - Scope of the document
  - List of references which should include references on test plan, functional specification, test case specification, etc.
  - Definitions, acronyms and abbreviations used in Test Design Specification

- **Features to be Tested** identifies test items and describes features and combinations of features that are the object of this design specification. Reference on Functional Specification for each feature or combination of features should be included.

- **Approach Refinements** section describes the following:
  - Specific test techniques to be used for testing features or combinations of features
  - Types of testing which will be provided
  - Methods of analyzing test results
  - Test results reporting
  - Whether automation of test cases will be provided or not
  - Any other information which describes approach to testing

Test Design Specification Structure

- **Feature Pass/Fail Criteria** specifies the criteria to be used to determine whether the feature or feature combination has passed or failed

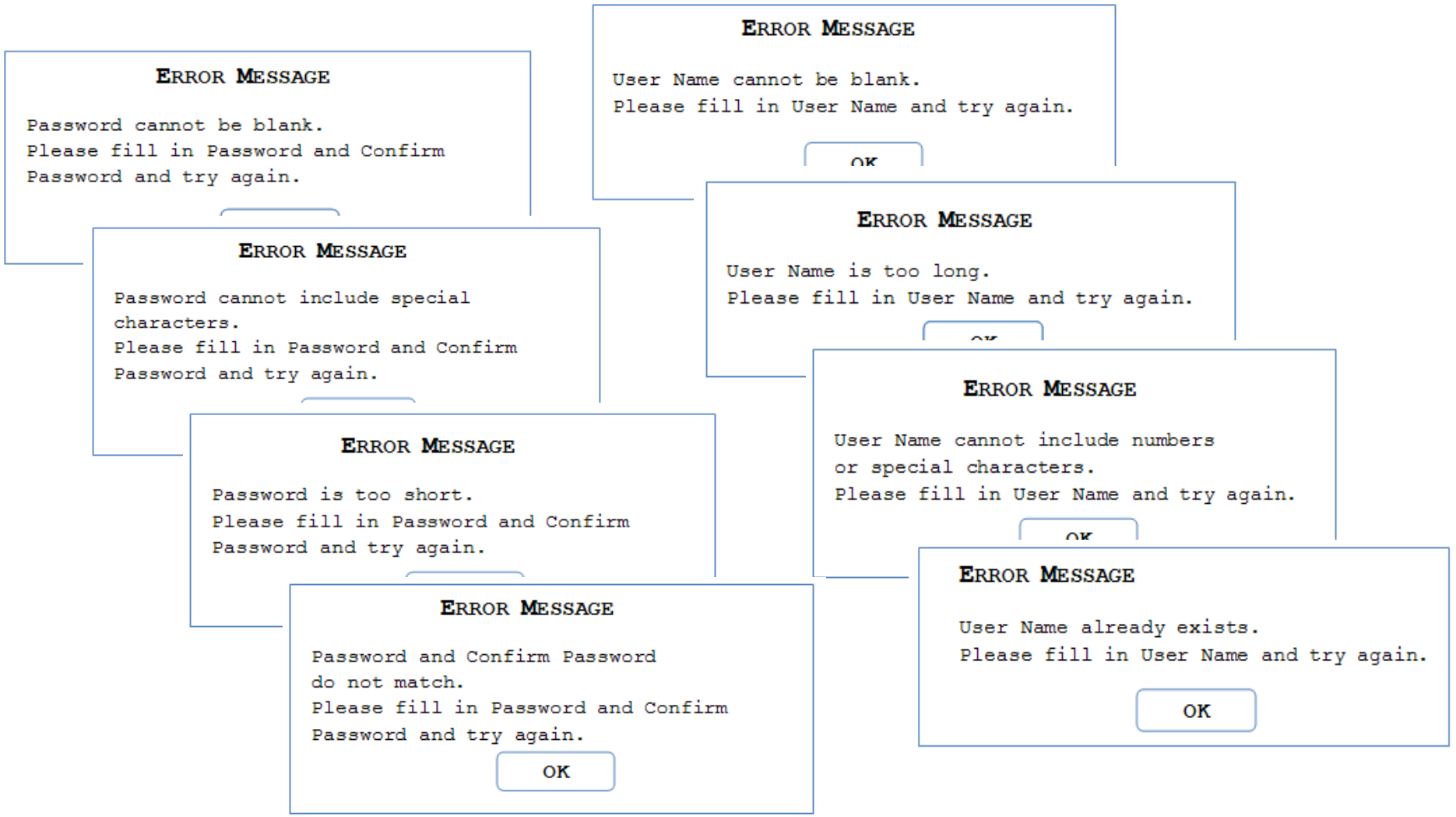
- **The following items can be considered as “pass / fail criteria”:**
  - Feature works according to stated requirements
  - Feature works correctly on the test platforms
  - Feature works correctly with other modules of application
  - All issues with High and Medium Priority will be verified and closed

- **Feature Pass/Fail Criteria** specifies the criteria to be used to determine whether the feature or feature combination has passed or failed

- **The following items can be considered as "pass / fail criteria"**:
  - Feature works according to stated requirements
  - Feature works correctly on the test platforms
  - Feature works correctly with other modules of application
  - All issues with High and Medium Priority will be verified and closed

Example

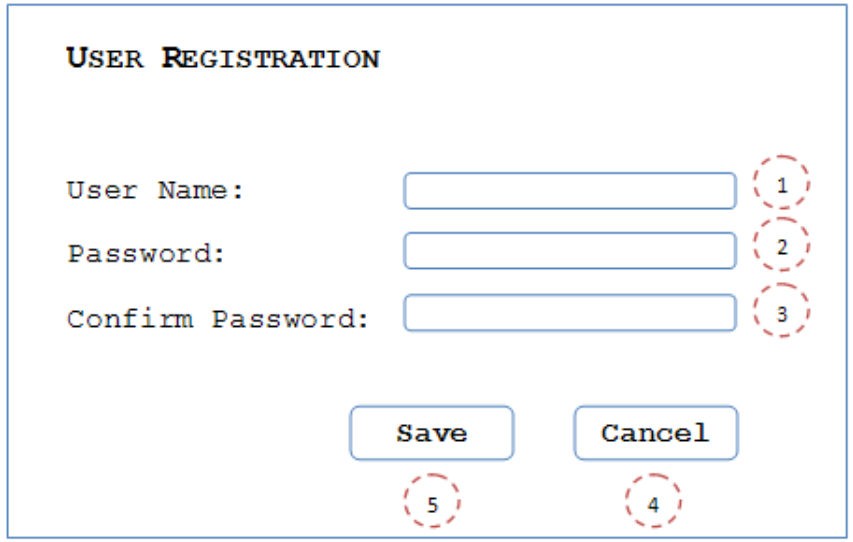
Requirements: Error Messages



Requirements: User Registration Page

- **Business Value:** I, as an Administrator user, should be able to create a simple user account to log in application.
- **Functional Requirements:** 'User Registration' page should contain three fields 'User Name', 'Password', 'Confirm Password' and two buttons – 'Save' and 'Cancel'.

- **Mock up:**



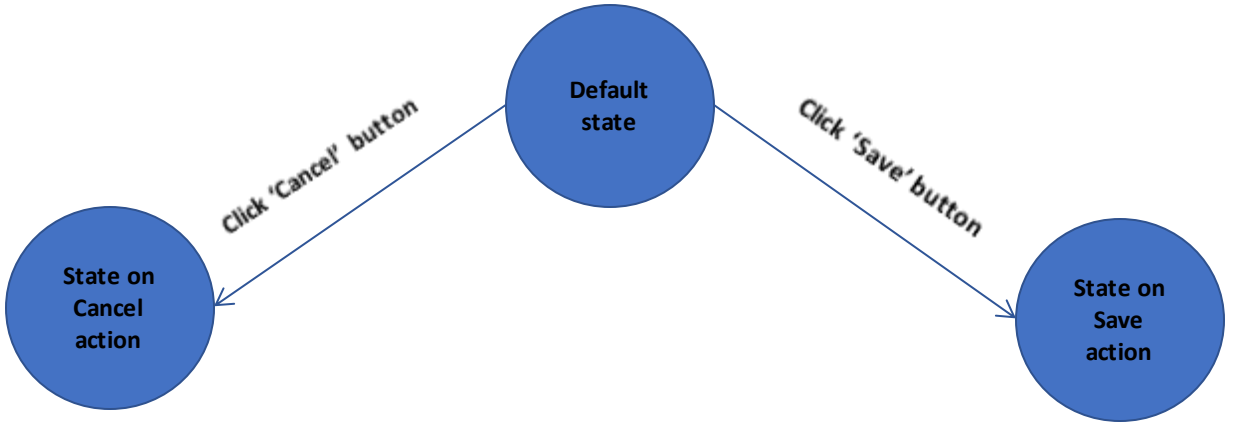
1. 'User Name' field is limited by 10 symbols and should contain letters of Latin alphabet only. 'User Name' field is empty by default. User Name should be unique in the system.
2. 'Password' field should be no less than 4 symbols long and should include only numbers and letters of Latin alphabet only. 'Password' field is empty by default.
3. 'Confirm Password' field should be equal to 'Password'. 'Confirm Password' field is empty by default.
4. 'Cancel' button cancels account creation and closes 'User Registration' page.
5. 'Save' button validates data entered into fields on 'User Registration' page and creates user account if entered data are correct; or shows error dialogs if validation fails. Validation should be provided in following order: User Name, Password, and Confirm Password.

Applying State Transition Technique

‘User Name’ field is empty by default. ‘Password’ field is empty by default. ‘Confirm Password’ field is empty by default.

‘Cancel’ button cancels account creation and closes ‘User Registration’ page.

‘Save’ button creates user account if entered data are correct.

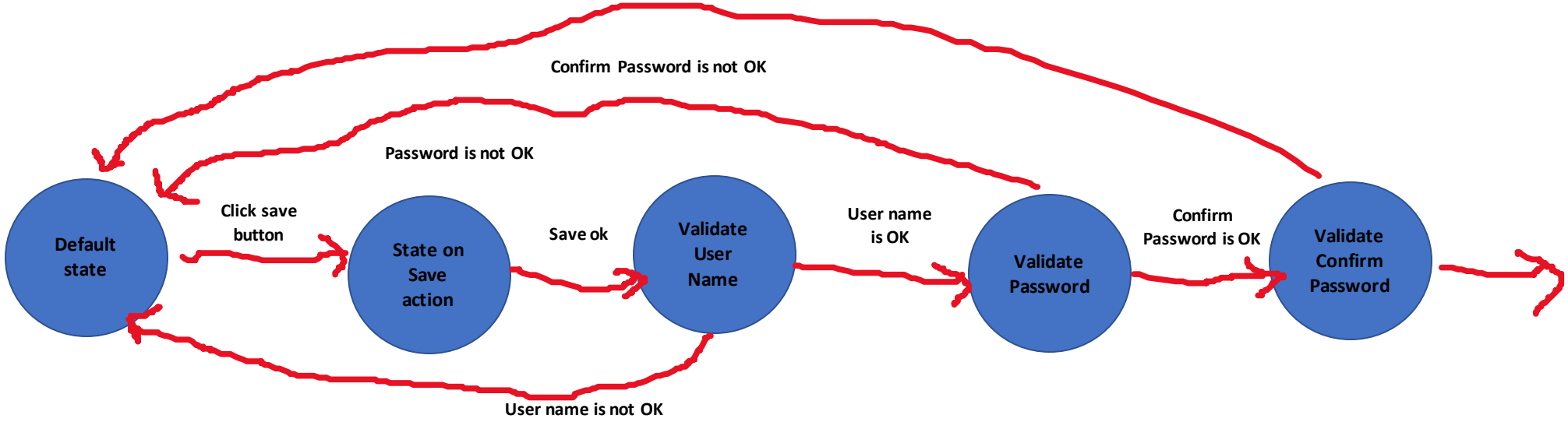


Applying State Transition Technique

Requirement	Test Name	Description
Default values	Default values on the 'User Registration' page	This test verifies that all fields on 'User Registration' page are blank by default
'Save' button functionality	Creating new user account and save	This test verifies that user account could be created if all fields on 'User Registration' page are filled with correct data; and 'User Registration' page is closed on save action
'Cancel' button functionality	Creating new user account and cancel	This test verifies that user account is not created after filling in fields on 'User Registration' page and canceling; and 'User Registration' page is closed on cancel action

Applying State Transition Technique

‘Save’ button validates data entered into fields on ‘User Registration’ page and creates user account if entered data are correct; or shows error dialogs if validation fails. Validation should be provided in following order: User Name, Password, and Confirm Password.



Applying BVA and EP Techniques

‘User Name’ field is limited by 10 symbols.

BVA: 0	1 10	11
Invalid Class	Valid Class	Invalid Class
EP: only 0	**1-10**	**11 and bigger**

‘User Name’ field should contain letters of Latin alphabet only.

Letters of Latin Alphabet	Numbers	Special Characters
Valid class	Invalid class	Invalid class
A-Z and a-z	**0-9**	**@, !, #, $, %, ^, &, *, (,), >, <, /, \, |, }, {,], [, ~,`, ‘, “, :, ;, etc.**

User Name should be unique in the system.

Test Item “User Registration”

Requirement	Test Name	Description
‘User Name’ field validation	Error dialog on saving user account with too long user name	This test verifies that error dialog appears while save action if user name length is too long: 1)boundary length – 11 characters 2)restricted length – more than 11 characters
	Error dialog on saving user account with blank ‘User Name’ field	This test verifies that error dialog appears while save action if ‘User Name’ field is blank
	Verify boundary length for user name	This test verifies that user account having user name with boundary length 1 or 10 could be created
	Error dialog on saving user account with wrong user name	This test verifies that error dialog appears while save action if ‘User Name’ field include: 1)special symbols; 2)numbers; 3)both
	Error dialog on saving already existing user account	This test verifies that error dialog appears while save action if user already exists in the system

Test Item “User Registration”

Requirement	Test Name	Description
‘Password’ field validation	Error dialog on saving user account with too short password	This test verifies that error dialog appears while save action if password length is too short: 1)boundary length – 3 characters 2)restricted length – less than 3 characters
	Error dialog on saving user account with blank ‘Password’ field	This test verifies that error dialog appears while save action if password is blank
	Verify boundary length for password	This test verifies that user account having password with boundary length 4 could be created
	Error dialog on saving user account with incorrect password	This test verifies that error dialog appears while save action if ‘Password’ field includes special symbols
‘Confirm Password’ field validation	Error dialog on saving user account with unequal password and confirm password	This test verifies that error dialog appears while save action if: 1)’Confirm Password’ field is blank 2)password and confirm password do not match

Next Task for Test Design Specification

- Choose features to be tested according to the number of your group member .*Eg: if you have three member , each member have to choose TWO features, so total will be 6 features*.
- For each of the assigned features, based on the requirements related to the features and using use case testing technique, you are required to identify :
  - The test condition
  - The test coverage
- For section 1B, you are required to design test case for the **USER** only. Choose features according to your group member. Same steps as no 1
- Test only functionality requirements related to that features

"How well we communicate is determined not by how well we say things, but how well we are understood."