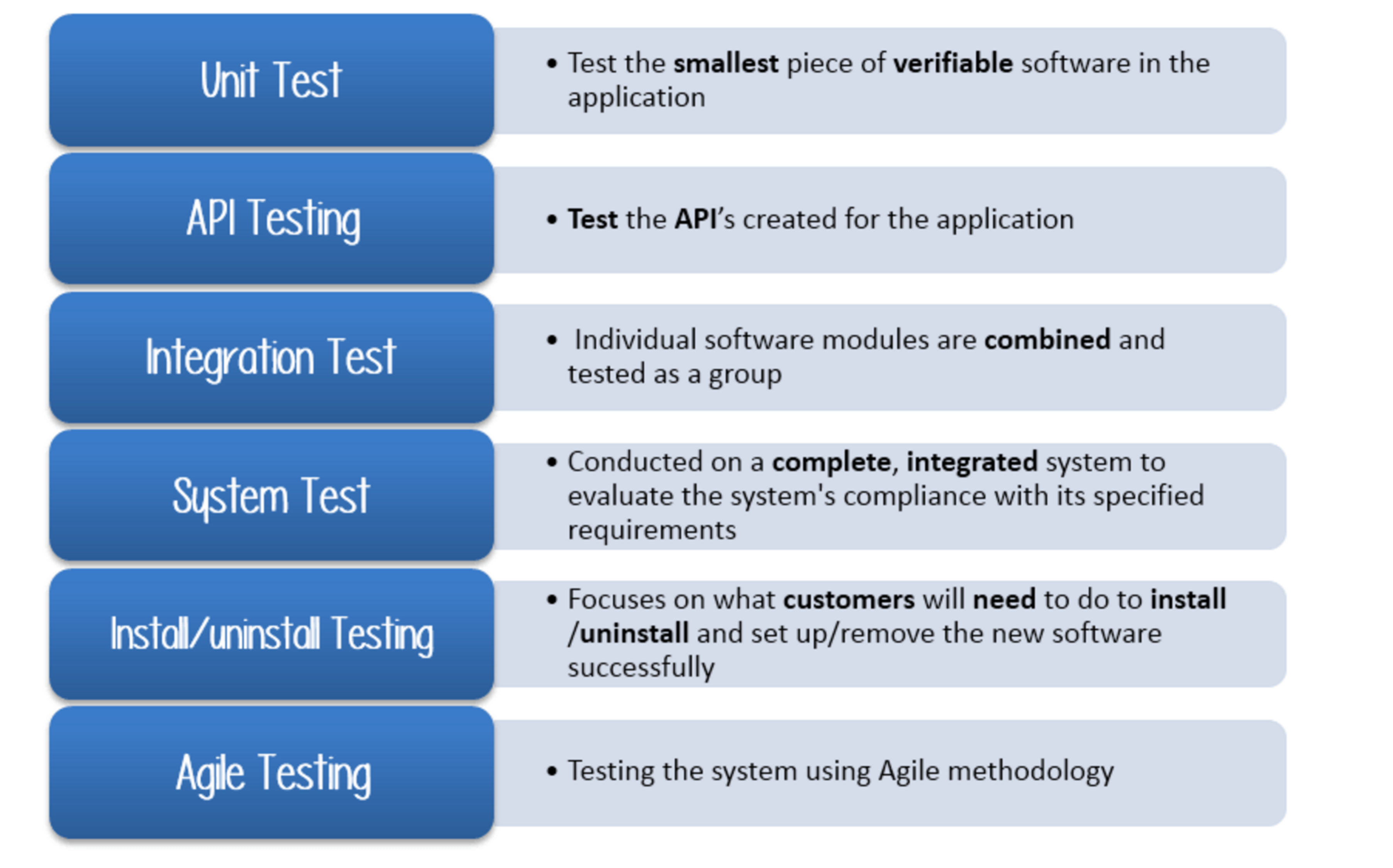
1.BENEFITS OF TESTING

* Test Plan helps us determine the **effort** needed to validate the quality of the application  under test
* Help people outside the test team such as developers, business managers, customers**understand** the details of testing.
* Test Plan **guides** our thinking. It is like a rule book, which needs to be followed.
* Important aspects like test estimation, test scope, test strategy are **documented** in Test Plan, so it can be reviewed by Management Team and re-used for other projects.

2.DIFFERENT TESTS



Person having the following skills is most ideal for performing software testing:

* Ability to **understand** customers point of view
* Strong **desire** for quality
* **Attention**to detail
* Good **cooperation**

3. DEFINE TEST OBJECTIVES

Test Objective is the overall goal and achievement of the test execution. The objective of the testing is finding as many software defects as possible; ensure that the software under test is **bug free** before release.

To define the test objectives, you should do 2 following steps

1. List all the software features (functionality, performance, GUI…) which may need to test.
2. Define the **target** or the **goal** of  the test based on  above features

It specifies the criteria that denote a **successful** completion of a test phase. The exit criteria are the targeted results of the test and are necessary before proceeding to the next phase of development. Example: **95%** of all critical test cases must pass.

Some methods of defining exit criteria are by specifying a targeted **run rate** and **pass rate**.

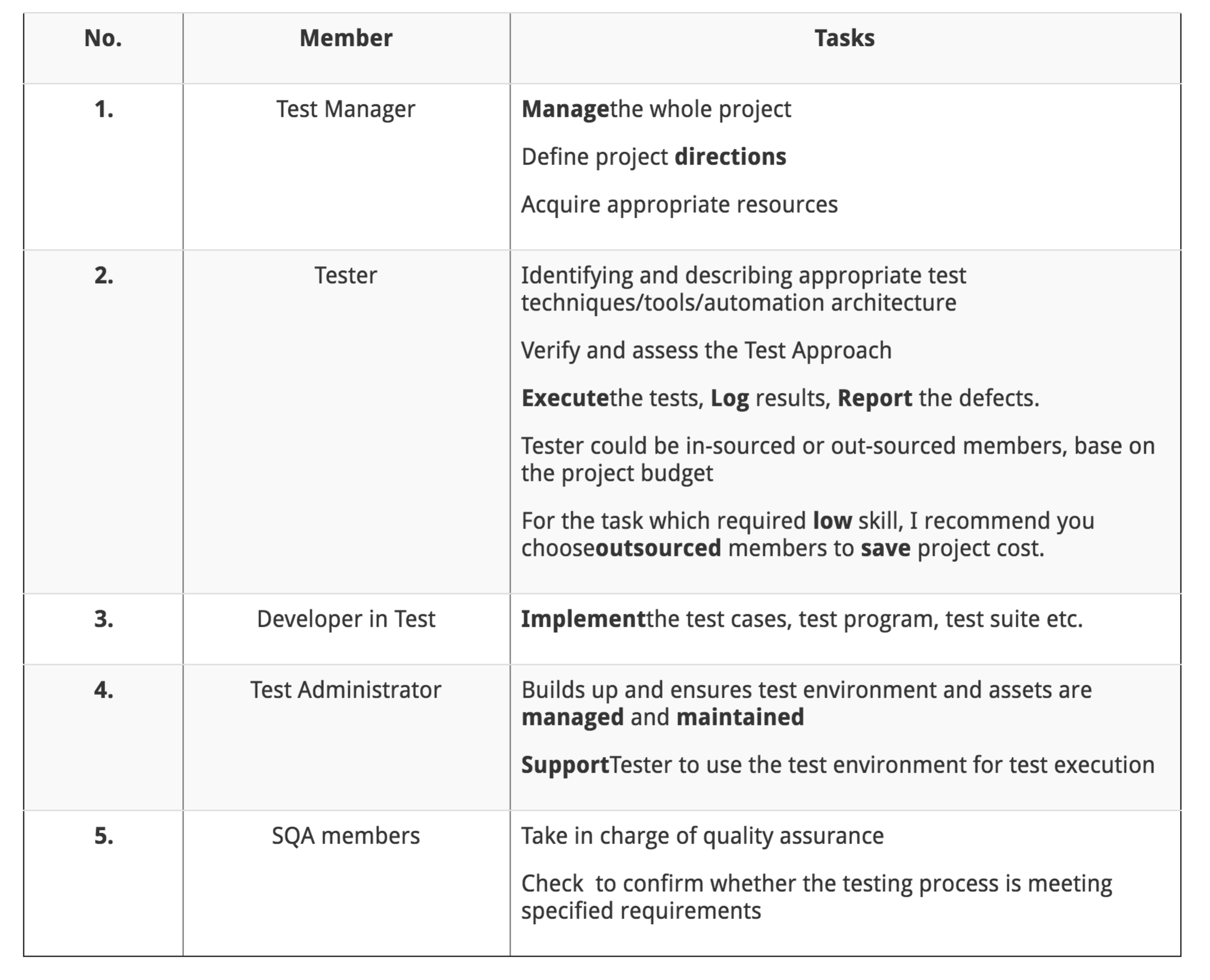
4. EXIT CRITERIA

* Run rate is ratio between **number test cases executed/total test cases** of test specification. For example, the test specification has total 120TCs, but the tester only executed 100 TCsàthe run rate is 100/120 = 0.83 (83%)
* Pass rate is ratio between **numbers test cases passed / test cases executed**. For example, in above 100 TCs executed, there’re 80TCs passedàthe pass rate is 80/100 = 0.8 (80%)

This data can be retrieved in Test Metric documents.

* **Run** rate is mandatory to be **100%**unless a clear reason is given.
* **Pass** rate is dependent on project scope, but **achieving high pass rate** is a goal.

**Example:**Your Team has already done the test executions. They report the test result to you, and they want you to confirm the **Exit Criteria.**

****

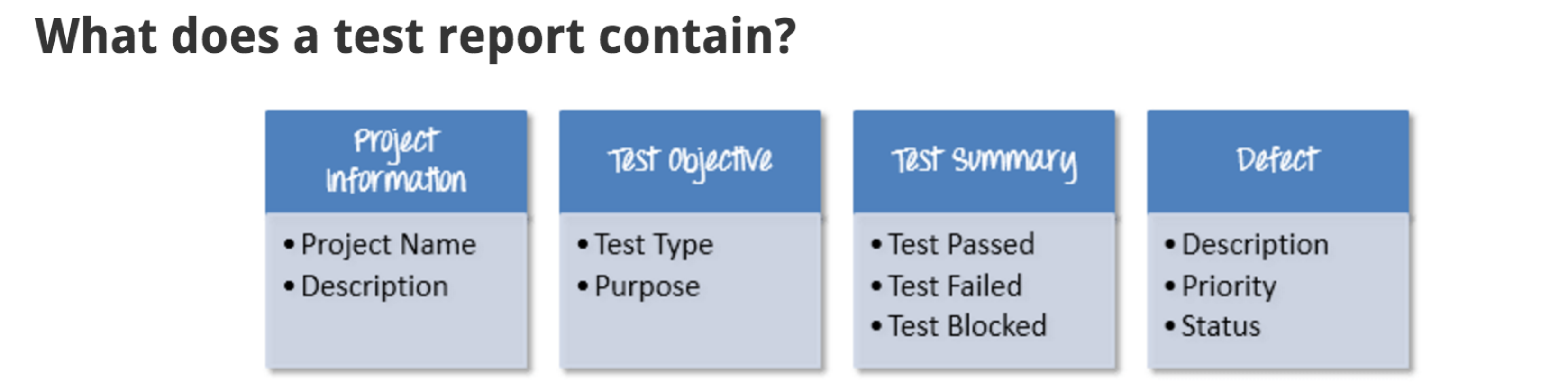
5. TEST REPORTS

Test Report is a document which contains

* A **summary** of test activities and final test results
* An **assessment** of how well the testing is performed

Based on the test report, the stakeholders can

* **Evaluate**the **quality** of the tested product
* Make a **decision** on the software release. For example, if the test report informs that there’re many defects remaining in the product, the stakeholder can delay the release until all the defects are fixed.



**Test Summary**

This section includes the summary of testing activity in general. Information detailed here includes

* The number of test cases executed
* The numbers of test cases pass
* The numbers of test cases fail
* Pass percentage
* Fail percentage
* Comments

This information should be displayed **visually** by using **color indicator**, **graph, and highlighted table**.

Take a look at [Test Report](http://www.guru99.com/images/TestManagement/Guru%20Test%20Report.xlsx) of the website Guru99 Bank to know more detail about Test report

**Defect**

One of the most important information in Test Report is defect. The report should contain following information

* Total number of bugs
* Status of bugs (open, closed, responding)
* Number of bugs open, resolved, closed
* Breakdown by severity and priority

Like test summary, you can include some simple metrics like defect density, % of fixed defects.



* **Detail**: You should provide a detailed description of the testing activity, show which testing you have performed. Do not put the abstract information into the report, because the reader will not understand what you said.
* **Clear:**All information in the test report should be **short** and **clearly** understandable.
* **Standard:**The Test Report should follow the **standard** template. It is easy for stakeholder to review and ensure the **consistency** between test reports in many projects.
* **Specific:**Do not write an essay about the project activity. Describe and summarize the test result specification and focus on the main point.

Defect Management

* **Defect\_ID** - Unique identification number for the defect.
* **Defect Description** - Detailed description of the defect including information about the module in which defect was found.
* **Version** - Version of the application in which defect was found.
* **Steps** - Detailed steps along with screenshots with which the developer can reproduce the defects.
* **Date Raised** - Date when the defect is raised
* **Reference**-  where in you Provide reference to the documents like . requirements, design, architecture or may be even screenshots of the error   to help understand the defect
* **Detected By** - Name/ID of the tester who raised the defect
* **Status** - Status of the defect , more on this later
* **Fixed by** - Name/ID of the developer who fixed it
* **Date Closed** - Date when the defect is closed
* **Severity** which describes the impact of the defect on the application
* **Priority** which is related to defect fixing urgency. Severity Priority could be High/Medium/Low based on the impact urgency at which the defect should be fixed respectively

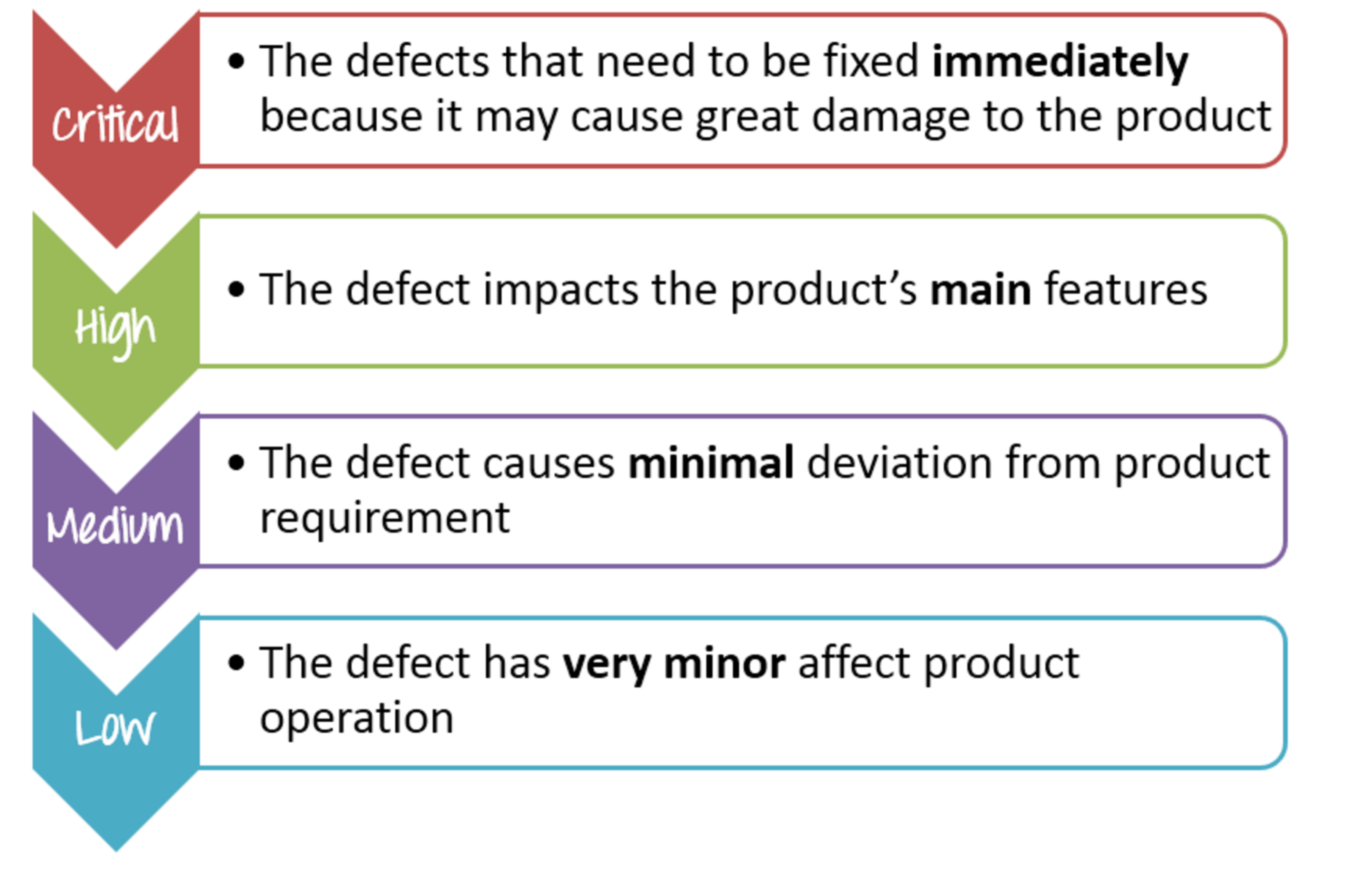
6. DEFECT MANAGEMENT PROCESS

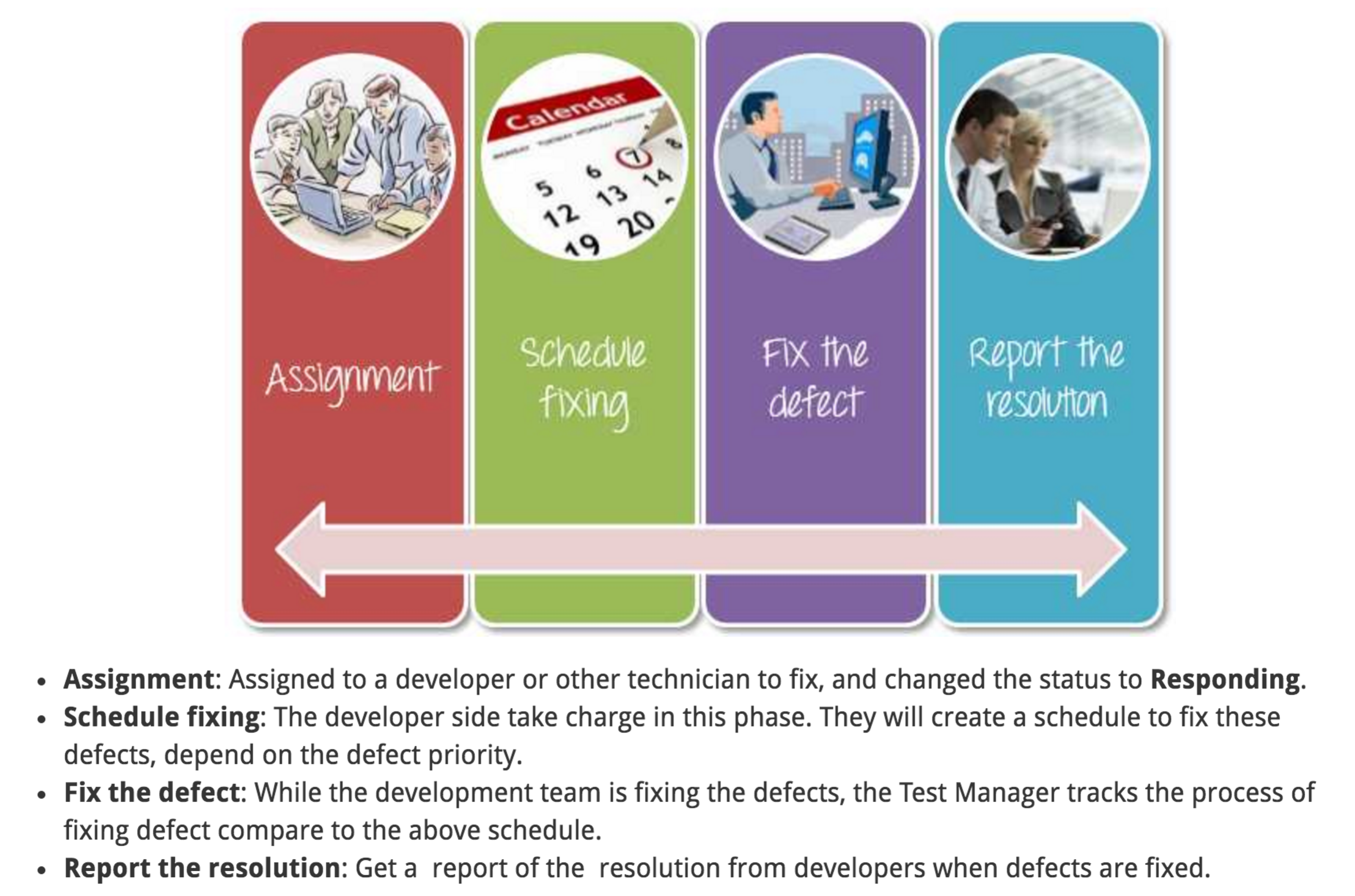


7.Discovery



8.Categorization





9.Measurement

