Testing/Quality Assurance Phase

The Quality Assurance Phase is a way of preventing mistakes and defects in deployed applications and avoiding problems when delivering them to customers. It is part of quality management focused on providing confidence that quality requirements will be fulfilled.

Unit Testing

UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output.

Integration Testing

**INTEGRATION TESTING** is a level of software **testing** where individual units are combined and **tested** as a group. The purpose of this level of **testing** is to expose faults in the interaction between **integrated** units. **Test** drivers and **test** stubs are used to assist in **Integration Testing**.

Alpha Testing

Alpha testing is the initial phase of validating whether a new product will perform as expected. Alpha tests are carried out early in the development process by internal staff and are followed up with [beta tests](https://whatis.techtarget.com/definition/beta-test), in which a sampling of the intended audience actually tries the product out.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Criteria** | **Poor** | **Fair** | **Good** | **Very Good** |
| Graphical User Interface (GUI) | | | | |
| Consistency (The user interface is of the same formatting style and icons throughout the system.) |  |  |  |  |
| Reusability (The system contains reusable GUI components such as familiar buttons, text and checkboxes, and other tools.) |  |  |  |  |
| Forgiveness and Tolerance (The interface displays message or confirmation prompts that would allow the users to undo or redo critical actions.) |  |  |  |  |
| Simplicity (The GUI design include simple GUI buttons, such as simple screens with clear, uncrowded messages.) |  |  |  |  |
| Readability (The interface has appropriate colors, font sizes, and styles that is convenient to the target users.) |  |  |  |  |
| Clarity (Displayed error, help, and warning messages are clear, concise, and as elementary as possible to assist user in operating the software.) |  |  |  |  |
| Flexibility (The system includes user preferences settings to allow changes, for example, increasing the font size.) |  |  |  |  |
| User-friendliness (The GUI design must be user-friendly, by providing helpful, courteous, and non-offending messages.) |  |  |  |  |
| System Performance | | | | |
| Conformance to the Requirements (The system effectively met all the identified features and/or requirements.) |  |  |  |  |
| Conformance to the Objectives (All specific objectives of the system are met by the program.) |  |  |  |  |
| Efficiency (The entire system functions efficiently. It doesn’t have delay in any transaction.) |  |  |  |  |
| Security (The system is secured. Login details are authenticated. Input parameters are ensured prior to the execution of the next transaction.) |  |  |  |  |
| Integrity (The software allows the registered user to have control over its own private information.) |  |  |  |  |
| Overall Impression (In general, the program or system is functional and useful.) |  |  |  |  |

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.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PROJECT IDENTIFICATION** | | | | | | | | | | | | |
| Project Name | | | | | | | | | | Date Created | | |
|  | | | | | | | | | |  | | |
| Project Sponsor/Owner | | | | | | | Project Manager | | | | | |
|  | | | | | | |  | | | | | |
| Project Adviser | | | | | | | Dean | | | | | |
|  | | | | | | |  | | | | | |
| **ACCEPTANCE CRITERIA TEST MATRIX** | | | | | | | | | | | | |
|  | | |  | | | **Critical** | | | **Test Results** | | |  |
| **Number** | | | **Acceptance Criterion Description** | | | **Yes** | | **No** | **Accept** | **Reject** | | **Comments** |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
|  | | |  | | |  | |  |  |  | |  |
| **OTHER CONSIDERATIONS** | | | | | | | | | | | | |
| Business Objectives – Did the software application meet the business objectives? | | | | | | | | | | | | |
| Yes | No | | |  | | | | | | | | |
| Does the software application system require any changes prior to installation? If so, please describe. | | | | | | | | | | | | |
| Yes | No | | |  | | | | | | | | |
| **APPROVAL** | | | | | | | | | | | | |
| Project Sponsor/  Project Owner | | | | | Signature | | | | | | Date | |
| Yes | | No | | |  | | | | | | | |