# Quality Attributes for Technical & Domain Testing

# Homework

1. You have a sample tests. Determine which characteristic of the software they are meant to test.

|  |  |
| --- | --- |
| * Test | * Characteristic |
| * Calculate the monthly discount for a client that is loyal for more than 3 years. Check if the system gives the same result. | * Functional requirements: Business Rules |
| * Check if the “zoom in” options work as they should. | * Non-functional requirements: Usability |
| * Check if you could access the system using SQL injection to log in instead of valid user credentials. | * Non-functional requirements: Security |
| * Check how the system behaves under Windows XP, Windows phone 7.5 and Android. | * Functional requirements: Portability |
| * Check if the new version of the application works with the old data base. | Functional requirements: Interoperability |
| * Check if the response time of the menu is less than 1.5 seconds. | Non-functional requirements: Performance |
| * Check if the application exports documents that can be opened with excel 2003. | Non-functional requirements: Data integrity |
| * Check if the system shows all the details needed to complete a contract with a client according to the specifications for that function. | Non-functional requirements: Serviceability/ Availability |
| * Check if the application works under the mentioned in the specification minimum hardware and software requirements. | Non-functional requirements: Performance/Stress |
| * Open the application and see a client’s profile. The services he/she uses. See if he/she contacted the call center or has sent a mail. | Non-functional requirements: Manageability |

*(10 positions X 2 points)* ***20 points***

1. Fill in the right term.

2**. S U I T A B I L I T Y**

3. **I N T E R O P E R A B I L I T Y**

4**. U S A B I L I T Y**

1. **S E C U R I T Y**

5. **H E U R I S T I C E V A L U A T I O N**

6. A C C E S S I B I L I T Y

1. Attributes of a software product that bear on its ability to prevent unauthorized access, whether accidental or deliberate, to programs and data
2. The capability of the software product to provide an appropriate set of functions for specified tasks and user objectives
3. The capability of the software product to interact with one or more specified components or systems
4. The capability of the software to be understood, learned, used, and attractive to the user when used under specified conditions
5. Static usability test technique to determine the compliance of a user interface with recognized usability principles
6. Tests the ability of users with particular requirements, restrictions, or disabilities to use the system

*(6 positions X 1 point)* ***6 points***

1. Fill in the right term.

**9.** **C O – E X I S T E N C E**

2**. R E L I A B I L I T Y**

**3.** **E F F I C I E N C Y**

**4.** **M A I N T A I N A B I L I T Y**

1. **A C CU R A C Y**

**5.** **P O R T A B I L I T Y**

**7.** **R E P L A C E A B I L I T Y**

**6.** **E X T E N S I B I L I T Y**

**10**. **R E L I A B I L I T Y**

**8. I N S T A L L A B I L I T Y**

**11**. **R E C O V E R A B L I T Y**

1. The capability of the software product to provide the right or agreed-upon results or effects with the needed degree of precision
2. The ability of the software product to perform its required functions under stated conditions for a specified period of time or for a specified number of operations
3. The capability of the software product to provide appropriate performance, relative to the amount of resources used under stated conditions
4. The ease with which a software product can be modified to correct defects, modified to meet new requirements, modified to make future maintenance easier, or adapted to a changed environment
5. The ease with which the software product can be transferred from one hardware or software environment to another
6. The capability to be adapted for different specified environments without applying actions other than those provided for that purpose
7. The capability to be used in place of another specified software product for the same purpose in the same environment
8. The capability to be installed in a specific environment. We will include uninstallability in this category
9. The capability to coexist with other independent software in a common environment sharing common resources
10. Deliberately subjecting a system to negative, stressful conditions
11. The system's ability to recover from some hardware or software failure in its environment

*(11 positions X 1 point)* ***11 points***

1. Fill in subcategories for each main category.

* Suitability
* Security
* Stability
* Understandability
* Interoperability
* Analyzability
* Installability
* Maturity (robustness)
* Recoverability
* Accuracy
* Adaptability
* Learnability
* Attractiveness
* Testability
* Fault tolerance
* Time behavior
* Operability
* Coexistence
* Resource utilization
* Replaceability

|  |
| --- |
| Functionality |
| Suitability |
| Testability |
| Fault tolerance |
| Time behavior |

|  |
| --- |
| Reliability |
| Security |
| Stability |
| Maturity (robustness) |

|  |
| --- |
| Usability |
| Understandability |
| Learnability |
| Attractiveness |
| Operability |

|  |
| --- |
| Efficiency |
| Recoverability |
| Accuracy |

|  |
| --- |
| Maintainability |
| An Analyzabilityalyzability |
| Adaptability |
| Replaceability |
|  |

|  |
| --- |
| Portability |
| Interoperability |
| Installability |
| Coexistence |
| Resource utilization |

*(20 positions X 1 points)* ***20 points***

1. Fill in some attributes for Technical Testing in the blank positions.

* Can it perform the required functions?

Capability

* Will it work well and resist failure in all required situations?

Recoverability

* How easy is it for a real user to use the product?

Usability

* How speedy and responsive is it?

Performance

* How easily can it be installed onto its target platform?

Installability

* How well does it work with external components & configurations?

Interoperability

* How effectively can the product be tested?

Testability

* How economical will it be to build, fix or enhance the product?

Suitability

* How economical will it be to port or reuse the technology elsewhere?

Portability

*(8 positions X 2 points)* ***16 points***

1. This is exercise about Functional vs. Non-Functional Testing. Your tast is to locate each sentence in the correct colmn below.

|  |  |
| --- | --- |
| Functional Testing | Non-Functional Testing |
| Focuses on what the system does | Should be performed after functional testing |
| Business requirements are the inputs to this testing | Focused on how the system does what it does |
| Tough to do manual testing | Easy to do manual testing |
| Using tools will be effective for this testing | Performance parameters like speed, scalability are inputs to this testing |
| This testing is executed first | Load Testing |
| Manual testing or automation tools can be used | Security Testing |
| Black Box testing | Compatibility Testing |
| Unit Testing | Installation Testing |
| Regression Testing | Penetration Testing |
| User Acceptance testing | Stress Testing |
| Sanity Testing | Performance Testing |
| Smoke Testing | Volume Testing |
|  | Integration Testing |
|  | Migration Testing  White box testing |

*(27 positions X 1 points)* ***27 points***