# <Software Title> - Software Validation Form

## 1. Information about the Software

|  |  |
| --- | --- |
| QMS ID | <ID> |
| Name | <Name> |
| Version | <x.x.x> |
| Location | <url> |
| Processes | <processes in which this tool is used> |

## 2. Intended Use and Use Context

Describe intended use and usage context (e.g. automation, testing, control, altering). Include technical and usage requirements that the system shall fulfill.

## 3. Quality Relevance

*Rate these aspects with yes (y) or no (n). If any of these aspects are rated as yes, the system is quality relevant and should be validated.*

| **Criterion** | **Y/N** |
| --- | --- |
| Is the system used in one or more processes that steer the QMS? |  |
| Could the conformity of the organization’s medical devices be affected if the system does not work according to its specifications? |  |
| Could risks arise for patients, users, third parties or the organization if the system does not work according to its specifications? |  |
| Does the software generate or manage data / records that are relevant to the QMS or medical device approval by authorities? |  |
| Is the software used to generate electronic signatures on documents or records required by the QMS and/or state authorities? |  |

## 4. General Assessment

### 4.1 Software Category

* Infrastructure software (e.g. operating systems, databases, office applications, antivirus, network management software) (GAMP category 1)
* Non-configurable software (GAMP category 3)
* Configurable software (GAMP category 4)
* Custom (self-developed) software (GAMP category 5)

### 4.2 Risk Assessment

**List of Risks:**

* <list of risks>

**List of Risk Mitigation Measures (if necessary):**

* <list possible risk mitigation measures>

### 4.3 Criticality and Review Schedule

*Refer to section 10 for descriptions of the criticality classifications. If a software is not highly critical and widely adopted / commonly used, it can be continuously re-validated during use.*

* **Low** (review upon major changes)
* **Moderate** (review every year)
* **High** (review every 6 months)

## 5. Validation Plan

### 5.1 Participants

| Role | Name | Task(s) |
| --- | --- | --- |
|  |  |  |

### 5.2 Test Environment

* Software tool accessed with <Windows 10 20H2 on Google Chrome 88.0.4324.150>
* Reference User Manual

### 5.3 Testing Procedure

* Run software system on sample data

## 6. Validation Report and Requirements

### 6.1 Acceptance Criteria

The software is approved for use if it is validated successfully and works as expected.

### 6.2 Validation of Usage Requirements

| ID | Expected | Result | Pass? |
| --- | --- | --- | --- |
| U1 | e.g. “A radiologist can log in with their email and password.” | “Login with correct email and password grants access to the annotation tool.” | yes |
|  |  |  |  |

### 6.3 Validation of Technical Requirements

| ID | Expected | Result | Pass? |
| --- | --- | --- | --- |
| T1 | e.g. “Execute correctly in the specified runtime (Google Chrome).” | “The application runs correctly in Google Chrome.” | yes |
|  |  |  |  |

### 6.4 Summary of Validation

| Type | Total | Pass | Fail |
| --- | --- | --- | --- |
| Usage Requirements | 1 | 1 | 0 |
| Technical Requirements | 1 | 1 | 0 |

### 6.5 Conclusion

Approving the software for use is recommended due to the acceptance criteria being fulfilled completely.

## 7. Proof of Validation

You can optionally insert screenshots for proof of validation. Strictly speaking, this is not a hard requirement by the standards but it’s nice to show when you’re being audited.

|  |  |
| --- | --- |
| U1 | <insert screenshot> |
| T1 | <insert screenshot> |

## 8. Approval and Release

| **Date of Approval** | **Name of Approver** |
| --- | --- |
| <date> | <name> |

## 9. History

| Date | Name | Activity |
| --- | --- | --- |
|  |  | <Initial Approval> |

## 10. Annex: Additional Information for Criticality Classification

**GAMP Implications**

* GAMP 5 always leads to “high” software criticality
* GAMP 4 always leads to a “high” or “moderate” software criticality, depending on further risk assessment in step 4.2
* GAMP 1 and 3 typically leads to a “low” or “moderate” software criticality, depending on further risk assessment in step 4.2

**Criticality High**

* A software failure can lead to harm, requiring medical intervention
* Software manages information relevant for vigilance purposes (e.g. customer safety information or recall actions)
* Software manages information that is quality-relevant and its loss would lead to an audit nonconformity

**Criticality Moderate**

* A software failure can lead to harm, however, not requiring medical intervention
* Software manages information that is quality-relevant and its loss would lead to an audit nonconformity

**Criticality Low**

* Software manages information that is quality-relevant and its loss would lead to an audit nonconformity

Template Copyright [openregulatory.com](https://openregulatory.com). See [template license](https://openregulatory.com/template-license).

Please don’t remove this notice even if you’ve modified contents of this template.