Scope

adidas uses JIRA for delivery planning, to track tasks and issues, to assign work, or follow team activity. As part of the Services, the Contractor shall ensure that the status of each entry in JIRA is maintained up to date. Source code, configuration and other build artifacts will be stored in Bitbucket. The other tasks and activities shall constitute a service contract according to Section 611 et seq. of the German Civil Code.

Before mentioned tasks and activities include for example:

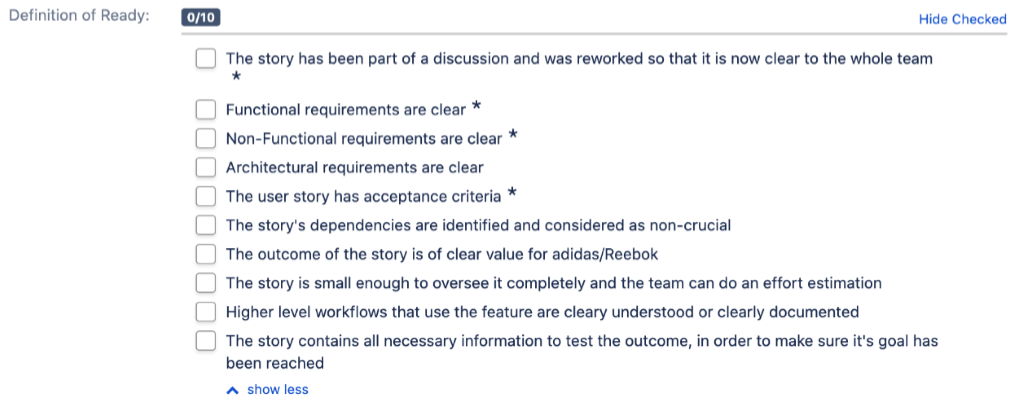
* *Support for the System Integration Testing (SIT), especially the resolving of Defects*
* *Support of the User Acceptance Test (UAT), especially the fixing of Bugs*
* *Provide adidas with a detailed delivery plan*
* *Maintain JIRA up to date with work in progress*
* *Complete all coding for approved functional and technical specifications*
* *Conduct basic functional testing*
* *Resolve any coding and configuration issues identified in testing*
* *Prepare and describe build procedure*
* *Present written status to adidas at bi-weekly meeting*
* *Proactively point out dependencies and potential problems*
* *Deployment to staging environment*
* *Continued maintenance of the Deliverables and the completion status*

Out of Scope

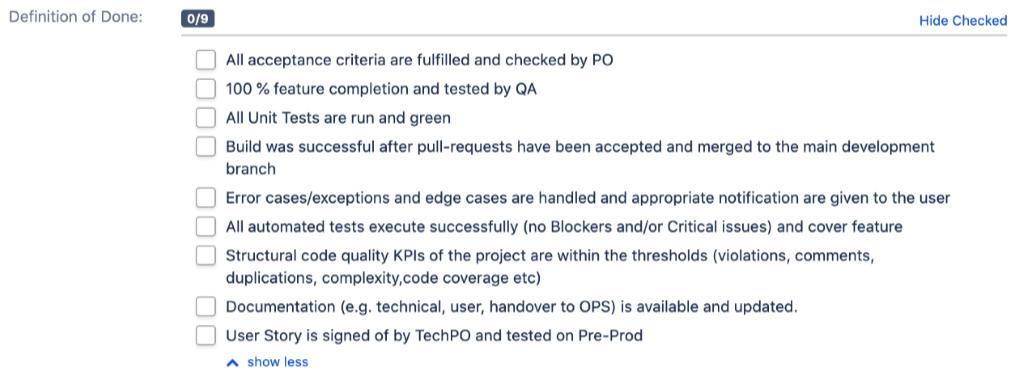
The following services are out of scope for this SOW:

* Keep system running activities (KSRs)
* Environment set up for Development / Testing and Stage
* Providing Go-live decisions, and approvals

Definition of Ready



Definition of Done



1. **Milestones and Timelines**

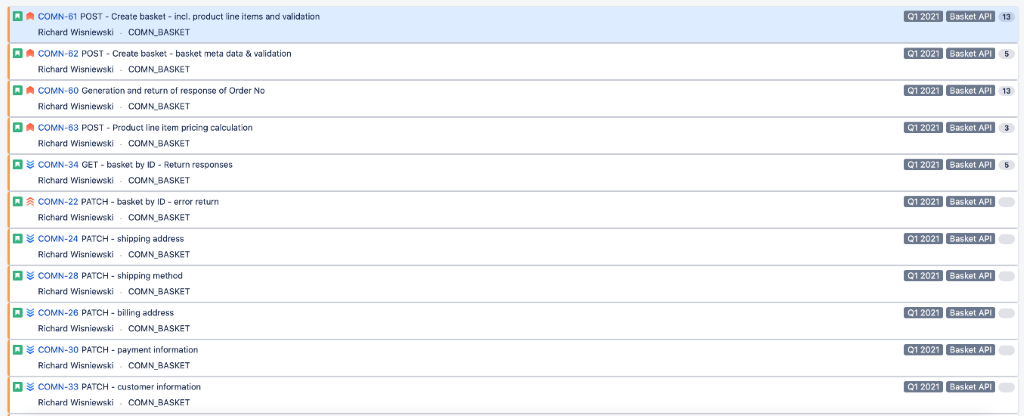
The Contractor will use all reasonable efforts to comply with the following Milestones and timelines.

* Deployment to “Staging Deployment” on 18.03.2021
* Deployment to “Public Staging DryRun” on 09.04.2021

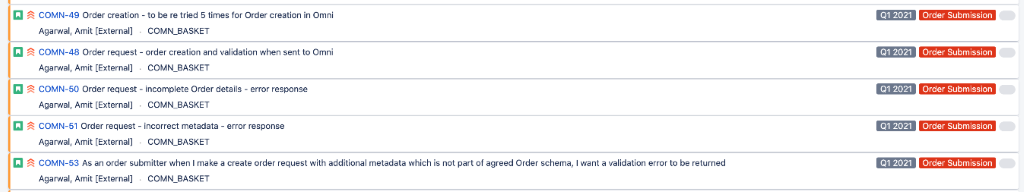
For each milestone there are two functional parts to be delivered: “Basket” and “Basket to Order interface”.

Back Log Items :

Basket API



Basket to order interface :



**Appendix E: Quality Requirements**

adidas will assess and measure the quality of the Deliverables based on the

* Testing Criteria (=acceptance criteria for each User Story), as defined by the individual product or project teams, and as documented in the relevant ticket took (e.g. JIRA), and
* minimum quality requirements described in this Appendix (**“Quality Requirements”**), which are relevant at all stages of the agile project, e.g. during the development, the testing as well as in the production environment.

In case of contradictions between the Testing Criteria and the Quality Requirements, the Testing Criteria will prevail. The Testing Criteria and Quality Requirements will be determined by standard tools and evaluated based on specific metrics or KPI’s as further specified below. If any Deliverable does not comply with any metric or KPI covered in this Appendix, the Deliverable will not be considered as first time right or as having passed the Sprint Testing. All Deliverables must comply with the Quality Requirements for adidas to give Acceptance to the Software.

The Contractor will provide the Deliverables in compliance with the following Quality Requirements:

1. Development Defect Leakage
2. Structural Code Quality
3. Technical Debt
4. Security KPIs

1. **Development Defect Leakage**

[Description of the KPI]

The Development Defect Leakage measures the number of Defects or Bugs reported in JIRA or a similar tool, which are either found during the Sprint Testing or after deployment of a Deliverable to the production environment.

[Minimum quality requirement]

If **SonarQube** is used, the minimum Quality Requirement is determined by the following formula, whereas the RCI must be above 95% for a Deliverable to pass the Sprint Testing.

Issue weight = blocker violations \* weight (10)

             + critical violations \* weight (5)

             + major violations \* weight (3)

             + minor violations \* weight (1)

             + info violations \* weight (0)

Rules Compliance Index (RCI) = max(1.0 - (Issue Weight / Lines of Code) \* 100, 0)

If the number of Defects is assessed with any other tool, then the number of Defects must be below the numbers specified in below table for the Deliverable to pass the Sprint Testing.

|  |  |
| --- | --- |
| **Severity** | **# of Defects** |
| Blocker | 1 |
| Critical | 4 |
| Major | 8 |
| Minor | 16 |

DEFECT SEVERITY DEFINITIONS

**Blocker**

* The Business user is unable to continue a successful business operation in the system unless the issue is resolved
* Complete ~~L~~loss ~~or partial loss~~ of key functionality
* Loss of functionality (no workaround)
* System crash
* Massive performance degradation
* Data corruption – unexpected data or file formats resulting from data altering functionality
* Unrecoverable loss of data important to business users i.e. transactional data
* Security violation as specified in the Software Requirement as part of this SoW
* System interfaces issues with downstream systems (Data transfer/update, Data Validation)

**Critical**

* Loss or partial loss of key functionality
* Operational error – errors affecting operation of system (process hang, user request time outs … )
* Data integrity – accuracy, consistency or completeness of data is affected
* Performance degradation - above thresholds agreed in non-functional requirements or ones causing system errors
* Usability issue / UI issues impacting business operations (e.g. usability /UI issues impacting data entry operations, pagination, usage of filter section…)

**Major**

* Partial loss of functionality of the software, but allows the user to continue proceeding normal business operations
* Usability / UI issues causing confusion due to inconsistency or ambiguity.

**Minor**

* The business user decides that the issue around this functionality is not vital to his use of the system
* Cosmetics (e.g. font face/ font size/ text alignment horizontal, vertical, centered/spacing/icon design…)
* Misspellings
* ~~Service requests~~

[Consequences]

If the minimum Quality Requirements are not met, the relevant Deliverable will not be accepted by adidas (with the consequences defined in Section 3.6 of the SOW). In general, all found Defects with the severity levels Blocker and Critical will be prioritized among other sprint tasks and must be fixed according the assigned priority.

1. **Structural Code Quality**

[Description of the KPI]

The Structural Code Quality relates to the amount of code violations, duplications, code complexity, and comments.

[Minimum quality requirement]

Each Deliverable must meet the criteria for “good” as defined by the following tables and any other KPI’s defined below.

**Code Violations:**

The amount of acceptable code violations depends on the size of the project. Therefore, issues are based on their severity, whereas Deliverables must have zero (0) blocker issues and zero (0) critical issues as well as a RCI above 98% to meet the minimum Quality Requirements.

**Code Comments:**

Measures the amount of code level documentation as reported by a standard quality measurement tool, whereas code comments for Deliverables must at least meet the criteria for “good” as defined by the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Performance/Quality | Poor | Average | Good | Excellent |
| Documented APIs | < 75% | 75 % – 85 % | 85 % – 99 % | > 99 % |
| Commented-out lines of code | >100 | 100 – 50 | <50 | 0 |

**Code Duplications:**

Measures the amount of duplicated code against total lines of code as reported by a standard quality measurement tool, whereas code duplications for Deliverables must at least meet the criteria for **“**good**”** as defined by the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Performance/Quality | Poor | Average | Good | Excellent |
| Violations | >12% | 7% - 12% | 5% - 7% | < 5% |

**Code Complexity:**

Measures the complexity level of code as reported by a standard quality measurement tool, whereas code complexity for Deliverables must at least meet the criteria for “good” as defined by the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Performance/Quality | Poor | Average | Good | Excellent |
| Complexity/method [avg] | > 7 | 4 - 7 | 3 -4 | < 3 |
| Complexity/class [avg] | > 25 | 12 - 25 | 10 - 12 | < 10 |
| Complexity/file [avg] (not JavaScript projects) | > 23 | 10 - 23 | 8 - 10 | < 8 |

**Unit Tests Coverage:**

Measures the amount of code that is covered by unit tests, which should be automated and reproducible. Unit tests line coverage for Deliverables must at least meet the criteria for “good”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Performance/Quality | Poor | Average | Good | Excellent |
| Line Coverage | < 65% | 65% - 80% | 80% - 90% | > 90% |

[Consequences]

Any Deliverables not meeting all “good” rated criteria does not meet the minimum Quality Requirements and will not be accepted by adidas (with the consequences defined in Section 3.6 of the SOW).

1. **Technical debt**

[Description of the KPI]

The technical debt is the effort to fix all code smells (meaning certain characteristics in the code which indicate a deeper problem), as calculated by a standard tool and measured in minutes.

[Minimum quality requirement]

The trend for technical debt always must be positive, meaning that the technical debt at the end of a Sprint must either decrease or stay even with the technical debt at the beginning of the Sprint.

[Consequences]

If the technical debt is increased, all Deliverables connected to that increase do not meet the minimum Quality Requirements and will not be accepted by adidas (with the consequences defined in Section 3.6 of the SOW).

1. **Security KPIs**

[Description of the KPI]

The Security KPI measures the quality of code and compliance level of the Deliverable as determined by security tools and procedures.

[Minimum quality requirement]

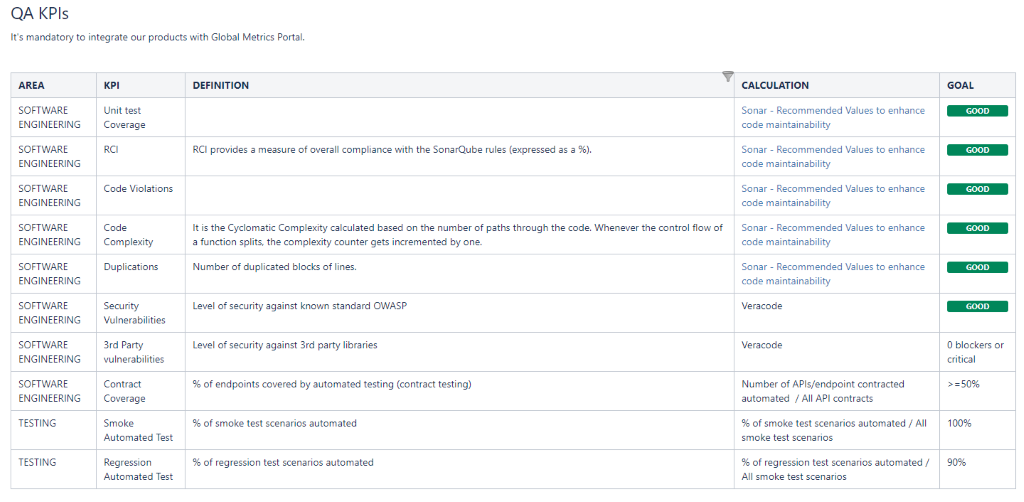
For each Deliverable, a minimum compliance level regarding both quantity and quality rated as “good”, as defined by below tables, must be met for the Deliverable to have met the minimum Quality Requirements. Considering any pre-existing security issues, the security compliance level should not decrease at any time.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metrics** | Security compliance | Poor | Average | Good |
| Quantity | Compliance level | < 85% | 85 % – 99 % | > 99 % |
| Quality | Vulnerability severity | CVSS (5-8) >30%  CVSS (8-9) <15 %  CVSS (10) <1% | CVSS (5-8) >20%  CVSS (8-9) <10 %  CVSS (10) 0% | CVSS (5-8) >10%  CVSS (8-9) <5 %  CVSS (10) 0% |

[Consequences]

Any Deliverables not meeting the minimum Quality Requirements will not be accepted by adidas (with the consequences defined in Section 3.6 of the SOW). Any security gaps or incompliance with security standards discovered in any Deliverable under this SOW must be fixed by the Contractor within agreed timelines.

**Quality Requirements**



Page Break