|  |
| --- |
| D:\Users\Mohandas\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Outlook\O8ZW7CZ1\company_logo.png |
| KPI |
| Quality Objectives & Baselines for Development Team |
|  |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Date | Remarks |
| 1.0 | K. S. Mathew | 18-06-2018 | Initial Version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |
| --- |
| KPI – Quality Objectives & Baselines for Development Team |

Table of Contents

[1 PURPOSE 3](#_Toc517086849)

[2 SCOPE 3](#_Toc517086850)

[3 ABBREVIATIONS / DEFINITIONS 3](#_Toc517086851)

[4 INPUTS 3](#_Toc517086852)

[5 KEY PERFORMANCE INDICATORS 3](#_Toc517086853)

[5.1 SCHEDULE VARIANCE 4](#_Toc517086854)

[5.2 EFFORT VARIANCE 4](#_Toc517086855)

[5.3 SCHEDULE PERFORMANCE INDEX (SPI) 4](#_Toc517086856)

[5.4 COST PERFORMANCE INDEX (CPI) 5](#_Toc517086857)

[5.5 QUALITY OF WORK (QOW) - DEVELOPERS 5](#_Toc517086858)

[5.6 QUALITY OF WORK (QOW) - TESTERS 6](#_Toc517086859)

[5.7 BUG DENSITY - MODULES 7](#_Toc517086860)

[5.8 BUGS - AGING 7](#_Toc517086861)

[5.9 REQUIREMENT STABILITY INDEX (RSI) 8](#_Toc517086862)

[6 QUALITY OBJECTIVES & BASELINE VALUES 8](#_Toc517086863)

[7 RESPONSIBILITIES 9](#_Toc517086864)

[8 QUALITY RECORDS 9](#_Toc517086865)

[9 ANNEXURE LIST 9](#_Toc517086866)

# PURPOSE

The purpose of this document is to define the Key Performance Indicators (KPI) for software development team of Indocosmo Systems Pvt. Ltd., herein after referred as ICS. The metrics collection & analysis templates for the KPIs as well as the organizational baseline values set for these metrics are also provided as part of this document. The metrics collection and analysis shall be done by the team and presented in management review meeting on every month. The baseline values to be re-defined and updated in the document at least once in a year.

# SCOPE

This document is applicable for all projects and team members in ICS.

# ABBREVIATIONS / Definitions

|  |  |
| --- | --- |
| **Abbreviation / Term** | **Description** |
| AC | Actual Cost |
| Cost Variance | Deviation from the approved budget of the project in % |
| CPI | Cost Performance Index |
| EV | Earned Value |
| ICS | Indocosmo Systems Pvt. Ltd. |
| PV | Planned Value |
| RSI | Requirement Stability Index |
| Schedule Variance | Deviation from the approved schedule of the project in % |
| SPI | Schedule Performance Index |
| QoW | Quality of Work |

# inputs

* Quality Policy and Quality Manual
* Procedures for Software Design and Development

# Key Performance INDICATORS

Key Performance Indicators (KPI) are used for assessing the performance of software development teams and the individual members of the team. The following sub sections define nine KPIs decided by ICS management for software development teams. These may also be considered as Quality Objectives of development teams. The templates for collecting and analyzing these metrics are provided as annexure of this document. Please refer ICS\_KPI\_Metrics\_Analysis\_Template. Baseline values for various metrics are provided in section 6 of this document. These shall be re-baselined, updated in this document and communicated to the software development teams once in every year.

## Schedule variance

The schedule variance is a measure of delay in delivery of a project, measured as a percentage of number of days delay w.r.t. total project duration. That is:

* Schedule Variance = (Actual End Date – Planned End Date)\*100 /

(Planned End Date – Start Date)

Schedule variance metrics shall be collected and analyzed as part of project closure and reported in project closure report as well as in monthly management review meeting. The present organization standard baseline value for schedule variance is given in section 6.

## effort variance

The effort variance is a measure of effort overrun (or decrease) in a project, measured as a percentage of number of person days (or hours) difference w.r.t. total estimated effort in person days (or hours). That is:

* Effort Variance = (Total Actual Effort – Total Estimated Effort)\*100 /

Total Estimated Effort

Effort variance metrics shall be collected and analyzed as part of project closure and reported in project closure report as well as in monthly management review meeting. The present organization standard baseline value for effort variance is given in section 6.

## Schedule performance index (SPI)

The schedule performance index (SPI) is based on the Earned Value Management technique in project management. This gives the schedule performance of the project at any time during the project execution based on the progress and status of various tasks/activities of the project. It is measured using the formula given below:

* Schedule Performance Index (SPI) = Earned Value (EV) / Planned Value (PV)

Earned Value (EV) is the sum of estimated effort of all tasks/activities completed as on date. Planned Value (PV) is the sum of estimated effort of all tasks/activities scheduled to be completed as on date. If all tasks (not more or less) scheduled as on date are completed fully, then both PV and EV will be same and SPI is 1, which is ideal. SPI will be less than 1, if all the tasks/activities scheduled to be completed as on date are not completed fully. SPI will be greater than 1, if more tasks/activities than scheduled to be completed as on date are completed. SPI less than 1 indicates the project is behind schedule and SPI greater than 1 indicates that the project is ahead of schedule, and SPI of 1 indicates that the project is on schedule.

Schedule metrics shall be collected and SPI to be calculated as part of weekly project status/progress review report and then reported in monthly management review meeting. The present organization standard baseline value for SPI is given in section 6.

## cost performance index (CPI)

The cost performance index (CPI) is based on the Earned Value Management technique in project management. This gives the cost performance of the project (which in turn is effort in software projects) at any time during the project execution based on the progress and status of various tasks/activities of the project. It is measured using the formula given below:

* Cost Performance Index (CPI) = Earned Value (EV) / Actual Cost (AC)

Earned Value (EV) is the sum of estimated effort of all tasks/activities completed as on date. Actual Cost (AC) is the sum of actual effort of all tasks/activities (taken from Time Sheets of employees) completed as on date. If all tasks are completed taking exactly the same effort as estimated, then both EV and AC will be same and CPI is 1, which is ideal. CPI will be less than 1, if sum of actual effort of all the tasks/activities completed as on date is more than the sum of estimated effort of those tasks. CPI will be greater than 1, if sum of actual effort of all tasks/activities completed is less than the sum of estimated effort of those tasks. CPI less than 1 indicates that the project is overrunning the budget/cost or effort and CPI greater than 1 indicates that the project is under budget or effort.

Effort metrics shall be collected and CPI to be calculated as part of weekly project status/progress review report and then reported in monthly management review meeting. The present organization standard baseline value for CPI is given in section 6.

## Quality of work (qoW) - Developers

The Quality of Work of Developers (QoW-Dev) is a measure of quality of the code developed by a developer, measured as a percentage (inverse) of total bugs in the code of the developer w.r.t. total bugs in the system/application of the project. The severity of the bugs also to be considered while calculating the quality of work. The critical & blocker bugs count is given 60% weightage, major bugs count is given 30% weightage and other bugs count is given 10% weightage. Quality of Work of Developers is measured using the formula:

* Quality of Work of Developers (QoW-Dev) = 100 – (% contribution of Critical & Blocker

Bugs + % contribution of Major Bugs +

% contribution of Other Bugs)

where

* + % contribution of Critical & Blocker Bugs = (Sum of critical & blocker bugs in developer’s code / Total critical & blocker bugs in the system/application) \* 100
  + % contribution of Major Bugs = (Sum of major bugs in developer’s code / Total major bugs in the system/application) \* 100
  + % contribution of Other Bugs = (Sum of other bugs in developer’s code / Total other bugs in the system/application) \* 100

Quality of Work of Developers (QoW-Dev) shall be collected and analyzed as part of project closure and reported in project closure report as well as in monthly management review meeting. This is a comparative measurement out of 100% and hence there is no specific baseline value.

## quality of work (QOW) - testers

The Quality of Work of Testers (QoW-Test) is a measure of the efficiency of the tester in detecting the bugs, measured as a percentage of bugs detected & reported by the tester w.r.t. total bugs reported in the system/application of the project. The severity of the bugs also to be considered while calculating the quality of work. The critical & blocker bugs count is given 60% weightage, major bugs count is given 30% weightage and other bugs count is given 10% weightage. Quality of Work of Testers is measured using the formula:

* Quality of Work of Testers (QoW-Test) = (% of Critical & Blocker Bugs detected +

% of Major Bugs detected +

% of Other Bugs detected)

where

* + % of Critical & Blocker Bugs detected = (Sum of critical & blocker bugs reported by the tester / Total critical & blocker bugs in the system/application) \* 100
  + % of Major Bugs detected = (Sum of major bugs reported by the tester / Total major bugs in the system/application) \* 100
  + % of Other Bugs detected = (Sum of other bugs reported by the tester / Total other bugs in the system/application) \* 100

Quality of Work of Testers (QoW-Test) shall be collected and analyzed as part of project closure and reported in project closure report as well as in monthly management review meeting. This is a comparative measurement out of 100% and hence there is no specific baseline value.

## bug density - Modules

The Bug Density of Modules is a measure of the intensity of bugs in the module, measured as a percentage of bugs detected & reported in the module w.r.t. total bugs reported in the system/application of the project. The severity of the bugs also to be considered while calculating the bug density. The critical & blocker bugs count is given 60% weightage, major bugs count is given 30% weightage and other bugs count is given 10% weightage. Bug density of a module is measured using the formula:

* Bug Density of a Module = (% of Critical & Blocker Bugs detected in the module +

% of Major Bugs detected in the module +

% of Other Bugs detected in the module)

where

* + % of Critical & Blocker Bugs detected = (Sum of critical & blocker bugs reported in the module / Total critical & blocker bugs in the system/application) \* 100
  + % of Major Bugs detected = (Sum of major bugs reported in the module / Total major bugs in the system/application) \* 100
  + % of Other Bugs detected = (Sum of other bugs reported in the module / Total other bugs in the system/application) \* 100

Bug Density of Modules shall be collected and analyzed as part of project closure and reported in project closure report as well as in monthly management review meeting. This is a comparative measurement out of 100% and hence there is no specific baseline value.

## bugs - aging

The Bugs Aging is a measure of how long a bug remains open without fixing after it is reported. It is measured in number of days using the formula:

* Bugs Aging = (Date of Bug Closure – Date of Bug Reporting)

Bugs Aging shall be collected and analyzed using the Bug Tracking System as part of weekly project status/progress review report and then reported in monthly management review meeting. The present organization standard baseline value for Bugs Aging is given in section 6.

## requirement stability index (RSI)

The Requirement Stability Index (RSI) is a measure of requirement changes in a project, and is measured as a percentage (inverse) of Number of Requirement Additions/Changes in the project w.r.t. Total Number of Original Requirements. It is measured using the formula:

* Requirement Stability Index (RSI) = 100 - (% of New Requirements added after SRS +

% of Requirements modified after SRS)

Where:

* + % of New Requirements added after SRS = (Count of New Requirements / Total

number of Original Requirements) \* 100

* + % of Requirements modified after SRS = (Count of Modified Requirements / Total

Number of Original Requirements) \* 100

Requirement Stability Index (RSI) shall be collected and analyzed as part of project closure and reported in project closure report as well as in monthly management review meeting. The present organization standard baseline value for Requirement Stability Index (RSI) is given in section 6.

# quality objectives & baseline values

The baseline values for the KPIs listed above shall be decided (first time based on industry standard figures) and then re-calculated or rebaselined by the organization once in every year based on the data of previous projects. The present values are given in the table below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Quality Objectives & Baseline Values** | | | | | |
| **Sl. No.** | **Objective** | **As On** | | | |
| **Jun 2018** | **2019** | **2020** | **2021** |
| 1 | Schedule Variance | +/- 5% |  |  |  |
| 2 | Effort Variance | +/- 10% |  |  |  |
| 3 | Schedule Performance Index (SPI) | 0.95-1.05 |  |  |  |
| 4 | Cost Performance Index (CPI) | 0.90-1.10 |  |  |  |
| 5 | Quality of Work (QoW) – Developers | Comparative figures out of 100% | | | |
| 6 | Quality of Work (QoW) – Testers |
| 7 | Bug Density – Modules |
| 8 | Bugs – Aging | 3 days Max |  |  |  |
| 9 | Requirement Stability Index (RSI) | 95 +/- 5% |  |  |  |

# responsibilities

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| CEO | Monitoring of Key Performance Indicators (KPI) during monthly review meeting and suggest actions to improve in case of poor performance of project teams and members. |
| IT Director | Advise CEO on actions to be taken to improve the performance in case of poor performance of project teams and members. |
| QA Manager | Coordinate preparation and consolidation of metrics by all PMs and present the metrics during monthly review meeting. |
| Project Manager | Collect and consolidate the project and team member metrics and prepare presentation for monthly review meeting. |

# quality records

* Project KPI Metrics presented in Monthly Management Review meeting

# annexure list

* ICS\_KPI\_Metrics\_Analysis\_template