

## Call for Bids

**Name of Project:** Metrics for Maturity Model  
**Deadline for Bids:** 28. February, 2020  
**Hand in Bids to:** Matthias Kunkel, ILIAS e.V., info@ilias.de

Dear All,

the Technical Board of the ILIAS society hereby calls for bids for the aforementioned project.

The quality of the software is one of the key concerns of the Technical Board (see: [https://docu.ilias.de/goto\\_docu\\_grp\\_5089.html](https://docu.ilias.de/goto_docu_grp_5089.html)). To ease the evaluation of the quality of some part of the software and to obtain some indications on which components might deserve a closer look, the Technical Board plans to propose a maturity model based on some set of metrics (such as #of unit tests, the presence of a readme and so forth, see list below). This project therefore targets to obtain a set of scripts to generate the values of the metrics in each component. Important, these metrics and the model based on them do not provide a comprehensive estimate of the overall quality of some component. They can only give some indications on which components might deserve a closer look due to some suspect value of some metric (e.g. no maintainer present, or no unit tests available). Therefore, they might give reason to look into some component and maybe propose some measures to improve parts of it. However, they are not meant to trigger direct consequences on their own.

Details of the project are outlined on the next page. Please hand in your bids and a corresponding statement of work by email according to the information in the header.

Bids will be judged by the offered price and the details outlined in the statement of work. We are looking forward to your submissions.

Best regards,

Matthias Kunkel

on behalf of the Technical Board of the ILIAS society

## Project Details

<b>Name of Project:</b>	Metrics for Maturity Model
<b>Latest date of delivery:</b>	1. Mai, 2020
<b>Goals and Benchmarks:</b>	The Goal of this project is to have an extendable set of scripts to assess the values of the metrics listed below in an automated fashion. This output should be easy to read and thus provide a tool to assess the maturity of ILIAS components based on those metrics.

The metrics are:

- Documentation
  - README present
  - ROADMAP present
- Bugs
  - Closed/Total Amount
  - Closed/New in the last year
- Tests
  - #of Automated Tests
  - #of Testrail Cases
- Code
  - Dependencies and Reverse Dependencies to other modules and services
  - LOC (Lines of Code)
  - Tokens
  - #of Classes
  - #of Interfaces/Abstract classes
  - #of Files
  - #of Functions (Using PHP-Keyword function, which includes methods)
  - Dicto Violations
  - #of interfaces/classes/traits with a name in a non-global namespace
  - #of tpl/html files
  - #of js/ts files
  - existence of locally included external libraries

- Activity
  - Is maintained
  - #of Contributors in the last 2 years
  - #of PRs in the last two years

**Deliverables:**

- For each of the metrics listed above, either a script or the manually generated value for each component (along with a comment explaining, why it would be too hard or impossible to generate this value by an automated script) should be provided.
- A script generating a report in the form of a csv file, containing all metrics that can be generated automatically, with the name of the metric as one dimension and the name of the Service, Module or Library (listed in src) in the other dimension.
- A md file describing how the script generating the reports can be run and extended, if automations for further values need to be added in a later point of time.
- Optional: A written proposal how these metrics can be combined or used to gain valuable insights into the maturity of components.

**General Conditions:**

- The scripts should be written in bash or php.
- The scripts along with the md file should be handed in as PR to the trunk of the official ILIAS repo.
- The files should be inside the new folder CI/Metrics.
- Mapping between the code repo and Mantis is done by general component names being used on both sides: As Mantis categories and as a property "belong\_to\_component" in the maintenance.json file of a component. Note, that this will be announced and done in the next few weeks.