

# **Multi-Level Healthcare Information Modelling (MLHIM)**

## **Project GOVERNANCE MODEL**

### **Table of Contents**

Introduction.....	1
1.0 ROLES AND RESPONSIBILITIES.....	1
1.1 Users.....	2
1.2 Contributors.....	2
1.3 Maintainers.....	2
1.4 Project Lead.....	2
2.0 CONTRIBUTING.....	3
2.1 Contributor's Agreement.....	3
2.2 Social Media.....	3
3.0 FEATURE REQUEST AND BUG TRACKING.....	3
4.0 DOCUMENTATION.....	4
5.0 GitHub Repositories.....	4

# **Multi-Level Healthcare Information Modelling (MLHIM)**

## **Project GOVERNANCE MODEL**

### **Introduction**

Ensuring that the MLHIM project is developed by the community in a structured and transparent way requires a certain degree of governance. The following rules and processes have been adopted to strike a balance between encouraging anyone to make a contribution at any time and maintaining a high level of quality in the project artefacts.

The current management model for the MLHIM project is often referred to as the *Benevolent Dictator* model. It is the same one used by other open projects such as the Linux kernel and the Python programming language as well as others. See: [https://en.wikipedia.org/wiki/Benevolent\\_dictator\\_for\\_life](https://en.wikipedia.org/wiki/Benevolent_dictator_for_life) As the original creator of the MLHIM specifications and reference implementation; Timothy (Tim) W. Cook is the MLHIM BDFL. Of course without contributors and users there wouldn't be a purpose for MLHIM and therefore the project is openly licensed and highly democratic.

### **1.0 ROLES AND RESPONSIBILITIES**

There are quite a few ways to participate in the MLHIM project and community, and not all of them involve contributing source code to the project. Simply participating in social media forums, or filing bug reports or enhancement requests on GitHub or contributing to graphics and documentation maintenance are incredibly valuable forms of participation.

Breaking down the forms of participation in the MLHIM project into a set of roles, they look something like this: Users, Contributors, Maintainers and Project Leads.

#### **1.1 Users**

Users are the people who download and use the MLHIM reference model or CCD-Gen artifacts. Users are using the software, reporting bugs, making feature requests and suggestions. This is by far the most important category of people. Without users, there is no reason for the project.

How to become one: Implement MLHIM in your software and use it to create interoperability layers.

#### **1.2 Contributors**

Contributors are individuals who contribute to the MLHIM reference model and CCD design but do not have write access to the GitHub repository. Contributions can be in the form of changes to code or documentation (via a GitHub fork), or bug reports, but

# **Multi-Level Healthcare Information Modelling (MLHIM)**

## **Project GOVERNANCE MODEL**

could also include web site content like articles, FAQs, or screen shots.

How to become one: Contribute in any of the ways described above.

Integration of a Contributors' submissions is at the discretion of the project maintainer, but this is an iterative, communicative process. Note that for code to be integrated, a completed Contribution Licence Agreement (CLA) is required from each contributor or organisation. See the CLA information in Appendix A.

### **1.3 Maintainers**

Developers at the MLHIM Technical Development Unit (TDU) are responsible for merging contributors' patches, bug fixes, and new code from the fork of the source tree into the appropriate branch of the GitHub repository. Maintainers are responsible for making sure that these contributions do not break any shipped reference models.

The Maintainer is also responsible for checking that everyone who contributes code has submitted a CLA. See the CLA information below.

A Maintainer acts as the "gatekeeper" of the repository, helping to ensure quality across all reference model branches.

### **1.4 Project Lead**

The Project Lead is responsible for the overall direction and vision of a particular aspect of the overall MLHIM project i.e. documentation or over a specific reference model build. In addition to making sure that releases are planned correctly and completed on schedule the Project Lead decides who receives commit access to the source code repository. This ensures that the quality of the software and the associated documentation remains high and the conceptual integrity of the project is maintained.

The Project Leads for MLHIM are designated by Timothy W. Cook.

## **2.0 CONTRIBUTING**

If you are not an official member of a project team or do not have commit access to the source code repository then you can still make a contribution by submitting a patch.

A patch can be submitted by creating a fork of the MLHIM repository, creating a branch for your patch and submitting a pull request.

# **Multi-Level Healthcare Information Modelling (MLHIM)**

## **Project GOVERNANCE MODEL**

### **2.1 Contributor's Agreement**

In order to accept a patch we require that you first complete a Contributor's Agreement acknowledging certain terms and conditions for its use. Once this agreement has been completed and a patch is accepted then the differences will be applied to the original source code or documentation by the project team and committed to the source code repository on your behalf.

### **2.2 Social Media**

Instead of mailing lists, we use social media to communicate. The primary venue is the Google + community at:

<https://plus.google.com/u/0/communities/117032458601124539887> It is a private community to reduce SPAM. Just request to join and it is usually approved within a few hours.

Secondarily is on Facebook at <https://www.facebook.com/mlhim2> We also publish on Twitter but do not monitor the stream at <https://twitter.com/mlhim2>

### **3.0 FEATURE REQUEST AND BUG TRACKING**

Submitting bug reports and suggesting new features are important tasks to help improve the quality and functionality of MLHIM. Each of the MLHIM repositories has its own Issue tracker on GitHub. Please select the appropriate one, which is usually the 'specs' repository unless the Issue is about one of the demos or tools.

Before submitting a new bug, please search our issue tracker to check the bug hasn't already been reported. If it has, then you can add a comment.

The objective is to give enough information for someone with no prior knowledge of the bug to recreate the problem for themselves. Too little information may prevent them understanding when the bug occurs.

When submitting a request for a new feature make sure to explain it in as much detail as possible. Each new feature requires its own issue in the issue tracking system so that they can be addressed separately by developers.

### **4.0 DOCUMENTATION**

MLHIM documentation is provided in the 'specs' repository as well as being included in each release.

## **Multi-Level Healthcare Information Modelling (MLHIM)**

### **Project GOVERNANCE MODEL**

Since documentation needs to be developed in a structured way, to coincide with each reference model release, it is stored in the same version control repository. This means that to make a contribution to a document you need to submit a patch.

## **5.0 Source Repositories & Releases**

All MLHIM artefacts are stored in version control on GitHub.com.

**Specifications:** <https://github.com/mlhim/specs>

**Specification Releases:** <https://launchpad.net/mlhim-specs>

There are several tools and demos on GitHub and on Launchpad. These projects are in constant development and change fairly often. They can be found under the umbrella URLs <https://github.com/mlhim> and <https://launchpad.net/mlhim>

If you wish to create an open source project or demo then please announce it in the Google + community and the Project Leads will create a repository for you to work with.

The CCD-Gen <http://www.ccdgen.com> is not an open source project. It does however have a public repository for Issue management and a wiki at: <https://github.com/twcook/ccdgen-public>

# **Multi-Level Healthcare Information Modelling (MLHIM)**

## **Project GOVERNANCE MODEL**

### **Appendix A**

This is the Contributors License Agreement (CLA) for the MLHIM project. Each CLA is issued between the MLHIM project as represented by Timothy W. Cook and the individual contributor. We do not negotiate contributor agreements with organizations. Individual contributors are responsible for insuring that they are not in violation of their employment agreement(s) when contributing to the MLHIM project.

A CLA **is required** before any source code or documentation changes are accepted by the MLHIM project. A CLA **is not required** for filing bug reports or enhancement suggestions.

This agreement is for your protection as a Contributor as well as the protection of the MLHIM project and users of its products; it does not change your rights to use your own Contributions for any other purpose.

By signing this document:

You confirm that you have an active account on <http://www.GitHub.com>

You represent that each of Your Contributions is Your original creation. You represent that Your Contribution submissions include complete details of any third-party license or other restriction (including, but not limited to, related patents and trademarks) of which you are personally aware and which are associated with any part of Your Contributions.

You represent that you are not in violation of any employment agreement by making contributions to the MLHIM project.

You understand that your contribution(s) will be incorporated into the project and licensed by one or more licenses certified as being open source licenses by The Open Source Initiative <http://opensource.org/> or open content licenses from the Creative Commons <http://creativecommons.org/> as appropriate to the contribution.

You understand that you become a joint copyright holder of the MLHIM project artefacts and that your copyright in your work becomes a part of the shared copyright with all other project copyright holders.

You must submit this document by the method detailed here:

1. Print the entire MLHIM Governance document.
2. Complete the information form below including an original signature.
3. Scan the document into a single PNG formatted file. The scanned file size must be less than 100MB.
4. Save the file using the name format of MLHIM\_CLA\_[first]\_[last].png where [first] is replaced with your surname and [last] is replaced with your family name.<sup>1</sup>

---

<sup>1</sup> We realize that different cultures use different approaches to personal name recognition. However,

## **Multi-Level Healthcare Information Modelling (MLHIM)**

### **Project GOVERNANCE MODEL**

5. Digitally sign the document using a GPG key that is publicly available on the Internet.<sup>2</sup> You may optionally encrypt the file with the public key of [tim@mlhim.org](mailto:tim@mlhim.org)  
Key ID: BBB1634A Fingerprint: F262 3D03 A88B 4CA2 46D1 C765 2D68 6580  
BBB1 634A
6. Create a repository on GitHub.com named MLHIM\_CLA and upload the signed PNG document to that repository.
7. Email the link to this repository to Tim Cook tim at MLHIM org. Use a subject line of, MLHIM CLA.

Complete the section below in handwriting:

Date (ISO format YYYY-MM-DD): \_\_\_\_\_

Name: \_\_\_\_\_

GitHub ID: \_\_\_\_\_

By signing this document I certify that I have read and understand its complete contents.

Signature: \_\_\_\_\_

---

this gives us consistency in managing CLAs.

2 If you are not familiar with using Gnu GPG to sign documents and upload your public key then please see the documentation at <https://www.gnupg.org/gph/en/manual.html#AEN111> Use either public keyserver at <http://keyserver.pgp.com/> or <https://pgp.mit.edu/>