Low Level Design Documentation

Release 0.1

Shiv Shankar Dayal & Pavan Sharma

October 06, 2012

CONTENTS

1 Introduction **3**

2 Architecture **5**
2.1 Models 5
2.1.1 Questions & Answers 5

3 Indices and tables **9**

Contents:

CHAPTER

ONE

INTRODUCTION

This document describes the low level design of libreQA Q&A platform. Libraries used are CppCMS, and Mimetic. We also use Google, Facebook, Linkedin and Twitter jQuery APIs. The entire pages are sent as JSON data from server which is rendered in browser using jQuery. Some AJAX pre-loading is done to improve the responsiveness of application. We will use jQuery Javsscript framework. No HTML or CSS is directly written. Database used is MongoDB. Programming language used is C++ for server side and DB access. We will use CMake as build system and Kdevelop as our primary IDE. GNU/Linux 64-bit system will be used for development. Repository is located at https://github.com/shivshankardayal/libreqa.git Coding standards of CppCMS will be used for this development. Later we will use OpenMP and Open-MPI for parallel processing and distributed computing. For Mathematics we will use MathJAX and for syntax highlighting we will use syntax highlighter with TinyMCE.

Given below are the world wide web uris for these:

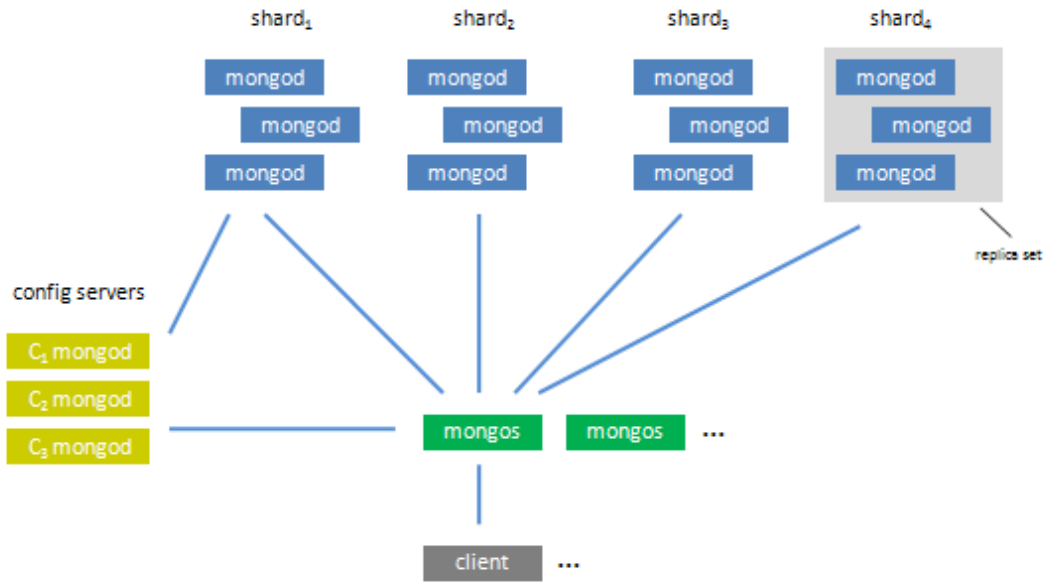
1. http://cppcms.com/
2. http://www.mongodb.org/
3. http://www.codesink.org/mimetic_mime_library.html
4. http://jquery.com/
5. http://www.tinymce.com/
6. http://www.mathjax.org/
7. https://github.com/efloti/plugin-mathjax-pour-tinymce
8. https://github.com/RichGuk/syntaxhl

CHAPTER

TWO

ARCHITECTURE

It is assumed that standard MongoDB sharding will be used as shown (the image was shamelessly copied from http://www.mongodb.org) below:



User can hit MongoDB's website to know about it.

2.1 Models

2.1.1 Questions & Answers

Folowing model(read collection) is in picture for questions and answers:

```
{
        Questions : {
                Question no. : Long Long,
                title        : String,
                Body             : html,
                tags             : Sorted List of Strings,
```

```
                view            : Integer,
                votes           : Integer,
                favorite        : Inetger,
                timestamp       : Timestamp,
                Followers       : List of users,
                Comments        :List of comments {
                        comment    : String,
                        timestamp  : TimeStamp,
                        commeneter : user,
                },
                answers : List of answers {
                        Answer id : Long Long,
                        answer    : String,
                        accepted  : Boolean,
                        votes     : Integer,
                        Comments  : List of comments {
                                comment    : String,
                                timestamp  : TimeStamp,
                                commeneter : user,
                        },
                },
        },
}
```

Following model(read collection) is for users:

```
{
        Users: {
                Name             : String,
                Email id         : String,
                URL                      : String,
                Date of Birth : Date,
                Karma            : Long Long,
                Skills           : Tags,
                Badges           : List of Badges {
                        Badge Name       : Name,
                        Date Acquired : Date,
                },
                Questions : List of Questions Asked {
                        Question URL : String,
                },
                Answers : List of Answers Given {
                        Question URL with answer ref: URL,
                }
        }
}
```

Following collection is for tags:

```
{
        Tags : {
                Name : String,
                Count : Integer,
                links : List of question for that tag {
                        link: URL
                }
        }
}
```

Following collection is for badges:

```
{
        Badges : {
                Users : List of people having that badge,
                Count : Long Long,
        }
}
```

CHAPTER

THREE

INDICES AND TABLES

- *genindex*