**ICGM-Developer Guide**

**Software Developer Guide**

**Mahesh K**

**Version: 1.0**

**Date: 08 June 2016**

# Document Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** | **Reviewer** |
| 08 June 2016 | 1.0 | Initial version of document | Mahesh K | Vivek Bendre |

Table of Contents

[1. Introduction 4](#_Toc452724298)

[1.1. Purpose 4](#_Toc452724299)

[1.2. Audience 4](#_Toc452724300)

[1.3. Prerequisition 4](#_Toc452724301)

[2. Classification 5](#_Toc452724302)

[2.1. Initializing the system. 5](#_Toc452724303)

[2.1.1. Database Configuration 5](#_Toc452724304)

[2.2. Working with the utility classes 5](#_Toc452724306)

[2.2.1. Age Class 5](#_Toc452724307)

2.2.2. AgeCalculator Class 5

[2.2.3. CHMModel Class 5](#_Toc452724307)

[2.2.4. CHMResponse Class 6](#_Toc452724307)

[2.2.5. PushDataToServer Class 6](#_Toc452724307)

[2.2.6. Back End Technology 6](#_Toc452724307)

[3. API References 7](#_Toc452724319)

[3.1. Mike-Phil android charts API 7](#_Toc452724320)

[3.2. Java SAX Parser. 7](#_Toc452724321)

[3.3. Android AssetManager. 7](#_Toc452724321)

# Introduction

This Developer Document provides documentation that will be used to aid in software maintenance by providing the details of how the app is built. This Document is brief documentation on source code for the ICGM project including other supporting requirement information.

## Purpose

**This guide provides information on the following:**

* **ICGM classification implemented with the exclusive features.**
* **ICGM application front end and back end.**

In order to best understand these concepts, the guide should be read in full. The guide is not intended to be used as a training or user guide.

## Audience

This guide is intended for developers who are using or extending the ICGM application as part of product development or a customer implementation.

## Prerequisition

It is assumed that the reader is familiar with the basic concepts of Java and android development. To better understand the topics covered in this guide, it is suggested that the reader has previously read the top level document.

# Classification

## Initializing the system.

### Database Configuration

Database Type: Android SQLite Database.

Package com.goalsr.kidsgrowth.kidsgrowthcharts.db describes the classes related to Database configuration. DBFeedReaderContract.class describes the Database tables and columns. DataBaseHelper.class describe database creation methods.

DB\_PATH: Path of the database.

DB\_NAME: Name of the database.

Tables: Doctors, Patients, Visits

Methods:

* createDataBase(String deviceId): Creates empty database on the system. deviceId is parameter used to save the device id.
* checkDataBase(): Check if the database already exist to avoid re-copying the file each time you open the application.
* copyDataBase(String deviceId): Copies your database from your local assets-folder to the just created empty database in the system folder, from where it can be accessed and handled.
* openDataBase(): Open the database to read and write from and to database respectively.

## Working with the utility classes

The util package provides a set of classes that provide predefined functionalities that can be reused. Following are the utility classes used in ICGM project.

### Age Class

Age class is used to create an Object which holds the age of patient in terms Days, Months and Years.

### AgeCalculator Class

AgeCalculator class is used to calculate the age of patients based visit date and date of birth.

Method: Age calculateAge(Date birthDate, Date visitdate): Returns an Age object based on two dates Birth date and visit date.

### CHMModel Class

CHMModel Class is used to hold an object of Doctor and Patient details. CHMModel class is used in pushing data to server as an ArrayList.

### CHMResponse Class

CHMResponse class is used hold the response of an Web server API call. CHMResponse has status, message, response\_code, record\_id details.

### PushDatToServer Class

PushDataToServer class describes the fetching of records from database and push it to server using rest template API.

Based on sync\_enabled value on the app side, the data sync will happen if the sync\_enabled value is true.

Web service API request contains an ArrayList(JSON ARRAY) of CHMModel.

Methods:

syncData() method connects with database and fetches the records from database and store the records in an arraylist. This method internally calls the sendToService() method if the sync\_enabled value is true.

### Back End Technology: PHP and MySQL Database

Used Code Igniter framework for backend implementation. Getting data from the application and pushing it to MySQL database and returning appropriate response.

**HealthCare Class of type Controller:**

Method : getrecords\_get()

Handles get request and returns all related records from database.

Method : insertPatient\_put()

Parameter : JSON object of fields and data

Handles put request and sends parameters to Model class to insert in database.

**HealthCareModel Class of type Model:**

Method: getList()

Interacts with database and gets all related records from db and sends to controller.

Method: insert()

Parameter: Associative array of fields and data.

Inserts all received data to database

# API References

## Mike-Phil android charts API:

Mike Phil android chart is a powerful & easy to use chart library for Android.

## Java SAX Parser

SAX (the Simple API for XML) is an event-based parser for xml documents.

## Android AssetManager

Android Asset Manager is repository for all the XML files.