Software DESIGN Document (SDD)



*Approvals:*

|  |  |  |  |
| --- | --- | --- | --- |
| Team Leader, Interface Specialist | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Alex Laird | Date: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Programming Specialist | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Ryan Morehart | Date: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Statistics and R Programmer | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Andrew Sterling | Date | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

*Document History:*

10-26-2010 – Initial Draft

Contents

[1 Project Overview 4](#_Toc275810630)

[2 Design Introduction 5](#_Toc275810631)

[2.1 User interface 5](#_Toc275810632)

[2.2 Problem 5](#_Toc275810633)

[2.3 R 5](#_Toc275810634)

[2.4 LaTeX 5](#_Toc275810635)

[3 Design Viewpoints 6](#_Toc275810636)

[3.1 Architectural Description 6](#_Toc275810637)

[3.1.1 Data Flow Description 6](#_Toc275810638)

[3.1.2 Hierarchy 7](#_Toc275810639)

[3.2 Component Description 7](#_Toc275810640)

# 1 Project Overview

* Context

Students entering into a first-semester probability and statistics course are often required to perform complex operations using the statistical command-line library of R. Students are also often required to format their homework neatly using the LaTeX framework. Both R and LaTeX are cumbersome hurdles that certain students have difficulty getting over to even be able to grasp the content of the course. This application is intended to eliminate the need for students to learn LaTeX and R as in depth, offering them a graphical interface to do the same sorts of things.

* Application

The application gives the user a simplistic graphical interface that allows them to perform complex statistical calculations. Problem sets can be created, solved, and exported to a neatly formatted Portable Document Format (PDF) file. Problem sets can be created easily using the simple graphical wizard for a new problem. Calculations can easily be made by dragging and dropping statistical commands onto the user’s data set, and the solutions to those calculations can be tied back to certain parts of the problem. To the average user, the complex calculations and exporting code will not be shown, but this code will be available at the request of a more advanced user.

* User(s)

The program is specifically targeted for students in their first-semester of a probability and statistics course. The scope could reach beyond students to anyone needing rudimentary functionality for simple probability and statistics equations, specifically if the user is seeking a simple graphical interface and not wanting to work with the command-line environment of R.

* Purpose

To eliminate the LaTeX and R learning curve from a first-semester probability and statistics course, and to provide an ease-of-use graphical interface for R that allows users to manipulate a drag-and-drop interface to perform complex statistics calculations in a simple manner.

# 2 Design Introduction

The requirements of maRla naturally leads to three major modules: user interface, R statistical backend, and the LaTeX output backend. In addition, the design of maRla introduces a middle layer between the user interface and backend to ensure a proper abstraction of the high-level concepts of problems and datasets. All interactions between the user interface and the backend flow though this middle layer.

The following modules are currently planned for maRla, most corresponding to a Java package.

## 2.1 User interface

This module comprises all user-facing components of the program. No interactions with the user should occur outside this code.

## 2.2 Problem

The middle layer of maRla, a Problem represents a logical statistical problem. Just like a homework problem, the Problem class contains data sets and problem statements. In the process of solving the problem, the user performs statistical operations on the data and indicates sequences of operations which solve parts of the problem.

## 2.3 R

The Java-R Interface Engine is a Java API to R. This module performs the low-level ties to this library. In addition, this module implements all of the statistical operations that a user may perform on data within maRla.

## 2.4 LaTeX

The LaTeX module exports parts of the problem and solution in LaTeX-formatted files. These allow the user to easily include them in their own LaTeX files. By keeping user-produced and maRla-produced LaTeX in separate files, the automatically created code may be updated without overwriting hand-produced work.

# 3 Design Viewpoints

## 3.1 Architectural Description

### 3.1.1 Data Flow Description

The diagram to the right illustrates the interactions described in the design introduction. As the diagram illustrates, the GUI drives all actions within maRla by working with the current Problem. The Problem creates and maintains DataSets as the GUI instructs and these DataSets maintain the Operations associated with them. Operations, when instructed, call out to R with the data from their parent DataSet and then save the values returned. The GUI receives data back by asking the main Problem to solve itself.



### 3.1.2 Hierarchy

* gui
  + Domain
  + MainFrame
    - ViewPanel
* problem
  + Problem
    - DataSet
      * DataColumn
      * Operation
    - SubProblem
* r
  + OperationMean
  + OperationNOP
  + OperationStdDev
  + OperationSummary
  + OperationSummation
  + RInterface
* resource
  + LoadSaveThread
  + Preferences

## 3.2 Component Description

* gui
  + ViewPanel – All interface components are contained within the view. Functionality for interactions directly related to the interface and the user are contained within this class, but anything indirectly related to the interface is abstracted out to the Domain class.
  + Domain – Encapsulates all user-interface logic that is indirectly related to the display. Specifically, relations between the front-end and the middle-layer are tied through the Domain class.
* problem
  + DataColumn – Contains the values in a column of a DataSet
  + DataSet – Table that may include one or more named columns of data
  + Operation – May be attached to DataSets or other Operations to perform a chain of statistical actions on data.
  + Problem – Holds a problem description and the named DataSets that are considered part of the problem.
  + SubProblem – Allows a chain of Operations/DataSets to be marked as the solution to part of the problem, allowing maRla to automatically export solutions to LaTeX correctly.
* r
  + OperationNOP – There are many derivatives of Operation, but OperationNOP provides a convenient test operation which simply echoes the data from the Operation or DataSet above it.
  + RInterface – Ties into the JRI
* resource
  + LoadSaveThread – Background thread that loads and saves maRla problems without locking up the interface. If the user preferences are set in such a way, the LoadSaveThread is responsible for periodically saving the current workspace automatically.
  + Preferences – Contains all user preferences specific to the interface and its interactions with R and LaTeX. The Preferences class is essentially and interface that contains an out function which spits all preferences to a preferences data file and can be built from that same string read from the data file.