Requirements Document

Student Performance Analysis

Shreyas Ugemuge sugemuge2014@my.fit.edu

Yaqeen AlKathiri yalkathiri2013@my.fit.edu

Mohammed AlHabsi malhabsi2013@my.fit.edu

Shiru Hou shou2015@my.fit.edu

Faculty Sponsor: Dr. Phillip Chan pkc@cs.fit.edu

February 22, 2017
v0.8

Contents

Revision history

Version	Date	Notes
v0.1	2/20/17	Format Skeleton
v0.5	2/21/17	sections 1.1 - 2.5 updated
v0.7	2/22/17	section 2.6 updated

1 Introduction

1.1 Purpose

The purpose of this document is to highlight the requirements in terms of features and functions that are expected from the final deliverable that will be a recommender system.

1.2 Scope

The final product will be an Academic Behaviour Recommendation System. The software will use the information gathered from click log files, syllabi and student grades to recommend for a certain student entry in that particular class the behaviours that were generally observed in that class which led to a good grade but were missing from that particular student's behaviour. The information about the behaviours is computed using a log file containing data about websites visited by the student during the course of the semester the class was taken in and does not consider what the student does beyond the web browser.

1.3 Definitions, Acronyms and Abbreviations

- ABRS - Academic Behaviour Recommendation System
- Behaviours - when used in the context of the data means the model in which the data is correlated to resemble a certain behaviour in real life. for instance, a student accessing an assignment link only 1 day before it was due may be a behaviour that may be interesting while correlating to grades.
- Log files - Excel files containing raw data i.e. 'clicks' from student over the course of the semester
- Syllabus - Document entailing details about a certain course. This includes test dates, assignment due dates etc.
- Grades - Lettered grade ranging from A - F where A is the best grade.
- CS - computer science (field of study)
- SWE - software engineering (field of study)

1.4 References

1. IEEE SRS standards guide - `http://www.math.uaa.alaska.edu/~afkjm/cs401/IEEE830.pdf`
2. IEEE SRS LaTeXGuide - `https://github.com/jpeisenbarth/SRS-Tex/blob/master/srs.tex`
3. Requirements document specifications - `http://cs.fit.edu/~pkc/classes/seniorProjects/document.html`

1.5 Overview

The entirety of this document is formatted and authored using the IEEE standards guide as well as the Requirements specified by the Instructor (See 1.4). The remainder of the document contains a detailed description of the project (See 2) and requirements (See 3) categorized by functional and non-functional.

2 Overall Description

2.1 Product Perspective

The product is standalone and is the entirety of the clients proposed system. The goal is for it to work with any kind of inputs that are of the same semantic and syntactical nature as the input provided by the client.

2.1.1 System interfaces

There are no anticipated special interfaces to be needed for the product to execute and the aim is to make it compatible with UNIX based and windows systems.

2.1.2 User Interfaces

1. A web browser for the GUI front end of the recommender system will be required
2. Capabilities to run and install python libraries

2.2 Product functions

The functions can be elaborated as follows:

1. Product will have a set of behaviours and methods to extract them given the required input
2. Product will be able to provide a report of behaviours identified for each student based on the data from log files and syllabi
3. The information as mentioned in the previous point will be correlated to the grades forming the idea of X behaviour correlates to a Good/Bad grade.
4. Provide a user interface that enables user to process the data, and use the information in the recommendation system
5. Recommender system will use the correlated data to compute the missing behaviours for any query student

2.3 User characteristics

There are various aspects that would influence as to who the end user is. Fundamentally the product is aimed at a single client that has very specific and unconventional raw data. The product could be useful for someone that uses the supporting documentation to prepare the input to be used with the product.
In summary any user apart from the intended client would have to have at least preliminary knowledge in the fields of CS or SWE.
Section 3 describe very relaxed requirements about the input which is due to the fact that the input that needs to be worked on is limited to files that will be organized in a set way that will be known as preliminary knowledge.

2.4 Constraints

The data is highly sensitive as it contains private data including grades and browser history. A strict privacy policy with the client eliminates the instance of having the files made available on any network.
With regards to the availability of said data, This kind of data cannot be obtained from a secondary source which limits the existence of any standards that may make the code more reusable.

2.5 Assumptions and dependencies

The code will be designed to work on conventional windows and unix based systems including MAC os. This is contingent on the fact that the computer has capabilities to install packages or have the required packages/tools pre installed.
makefiles and shell scripts will be designed to accommodate all platforms listed in 2.1.

2.6 Apportioning of requirements

This document will be revised significantly to include precise names of packages/tools used. The specifications have now been v 2.0 (Scheduled now for milestone 2) of the document because there are preliminary steps that need to be taken in order to make the choice for the optimal resources to be used for it.
The specific requirements in v 2.0 will further be classified by each behaviour. v 2.0 will also introduce detailed specification of the UI since that is delayed till later development stages.
A variety of tools are being researched for these purposes. Details can be located in the progress report documentation. Some packages hint that java may be required. If it becomes a requirement in the future the specific requirements may further be classified based on languages being used.

3 Specific Requirements

3.1 Functional Requirements

3.1.1 Shall be able to read data from syllabus for use with data mining for behaviours

3.1.1.1 Shall be able determine assignments and due dates and store for further use
3.1.1.2 Shall be able to determine Test Dates and store for further use
3.1.1.3 TBD ...

3.1.2 Shall have atleast 6 behaviors conceptualized and models to extract these from the data set

3.1.2.1 Determine the models that can be used on current input data
3.1.2.2 Depending on the fields obtained from the syllabus deploy appropriate data mining models to extract behaviour for each student
3.1.2.3 Shall Prepare the information for different tasks

3.1.2.3.1 Shall prepare human readable report for behaviours identified for query student
3.1.2.3.2 Shall prepare and organize data to be used for correlation purposes

3.1.3 Shall correlate Behaviours modelled with grades

3.1.3.1 Identify behaviours that show correlation to good grades (top 10%)
3.1.3.2 Identify behaviours that show correlation to bad grades (bottom 10%)
3.1.3.3 Identify and handle outliers
3.1.3.4 Shall Prepare the information

3.1.3.4.1 Prepare and organize information to be used by reccomendation syste
3.1.3.4.2 Shall Prepare information when needed for any query student in a human readable format for identified behaviours and achieved grade

3.1.4 Shall provide recommendation report for query student

3.1.4.1 Shall display a report in human readable format

3.1.4.1.1 Shall provide grade information and a statistical analysis of grade compared to the class
3.1.4.1.2 Shall provide list of behaviours that correlate to better grades for query student
3.1.4.1.3 Shall provide list of current behaviours to avoid for query student
3.1.4.1.4 Shall provide list of current behaviours that correlate to a good grade