***Project Name***: ***Automated Analysis of Selection Test Data*** Requirements Document (version 1.0)

Project: **Automated Analysis of Selection Test Data**

Date(s): **02/10/2018**

Prepared by: **Jayant Solanki**

Document status: **\_X\_** Draft \_\_ Proposed \_\_ Validated \_\_ Approved

**1. Introduction**

This document contains the system requirements for ***Automated Analysis of Selection Test Data***. These requirements have been derived from several sources, including:

* e-Yantra Robotics Competition Online Selection Test
  + [e-Yantra](http://www.e-yantra.org/): it is an initiative by IIT Bombay that aims to create the next generation of embedded systems engineers with a practical outlook to help provide practical solutions to some of the real world problems.
  + Every year e-Yantra organises Online Selection Test for the participants of e-Yantra Robotics Competition (eYRC). The purpose of the Selection Test is to select best participants for the later stages of the competition which can deliver on the challenges provided to them during those later stages.
  + The test data includes the details of the participants for the competition, responses of the test takers for the questions and the other subtle informations which will be provided later in this document.
* Machine Learning
  + This will be one of the important software component which will be used for the analysis of the date generated during the Online Selection Test.
  + One of the main task will be to predict the questions’ difficulty based upon the performance analysis of the test-takers.
* Web Application
  + This application will provide the interface for the user/question-master to learn about the important statistics and results of post machine learning analysis.

**1.1 Purpose of This Document**

This document is intended to guide development of ***Automated Analysis of Selection Test Data***. It will go through several stages during the course of the project:

1. **Draft:** The first version, or draft version, is compiled after requirements have been discovered, recorded, classified, and prioritized.
2. **Proposed:** The draft document is then proposed as a potential requirements specification for the project. The proposed document should be reviewed by several parties, who may comment on any requirements and any priorities, either to agree, to disagree, or to identify missing requirements. Readers include end-users, developers, project managers, and any other stakeholders. The document may be amended and reproposed several times before moving to the next stage.
3. **Validated:** Once the various stakeholders have agreed to the requirements in the document, it is considered validated.
4. **Approved:** The validated document is accepted by representatives of each party of stakeholders as an appropriate statement of requirements for the project. The developers then use the requirements document as a guide to implementation and to check the progress of the project as it develops.

**1.2 How to Use This Document**

We expect that this document will be used by people with different skill sets. This section explains which parts of this document should be reviewed by various types of readers.

**Types of Reader**

*In this section, list the different types of reader this document is aimed at. For example, Flash programmers, graphic designers, end-users, project managers, etc. For each type of reader, clearly state which sections are most pertinent to them, and which may be safely skipped.*

This document is aimed for readers which are:

* **Project-Manager:** Who wants to understand the working of the Selection Test, and how the quality of questions in a Selection Test can be maintained.
* **Question-Master:** Creates questions and get insights from previous questions to tag new questions correctly.
* **Machine Learning Enthusiast:** Provides a huge data-set to these users which can use it for feature generations, write effective algorithms and come up with better taggings.
* **Web-Developer**: Collaborate with machine learning programmers in coming up a web portal for utilising their training algorithms.
* **UX-Developer:** Design better interface for easy usability of the Web Portal based upon the client’s need.
* **Database Programmer:** Learn about the database schema in better depiction of the features and other vital data.

**Technical Background Required**

*Describe here the technical background needed to understand the document in general, and any particular expertise or understanding that is needed for specific sections.*

* The reader must be aware of the Online Test which are prevalent.
* The reader must know the basic usage of a computer web browser.
* The reader may know about supervised/unsupervised machine learning techniques.
* The reader may know about the database queries.
* The reader may know about the Full-Stack web development.
* The reader may have some experience in question designing.

**Overview Sections**

*List here the sections that should be read by someone who only wishes to gain an overall understanding of the project, or which should be read first before technical requirements are reviewed.*

**Reader-Specific Sections**

*In this section, name any parts of the document which are intended only for one or another of the reader types identified above, and which may therefore be skipped by other readers.*

**Section Order Dependencies**

*If readers will need to read certain sections in a specific order, note those sections here. Also point out any sections that may be read independently with no loss of understanding.*

**1.3 Scope of the Product**

*Include a brief narrative here which describes the product as you intend it to be realized. Use this section to define boundaries and set expectations.*

* **Core Features**
  + Identifications of those questions which remained same for all iterations of Selection Tests, that is in 2014, 2015, 2016 and 2017.
  + Identification of features in the Selection Test data. This step will be done in sync with the next step and will be altered frequently.
  + Development of Machine Learning Algorithm for predicting the tags of the questions.
* **Intermediate Features**:
  + Development of Laravel Framework based website with user login feature
  + Feature for adding new questions
  + Feature for showing retrospective tags for a particular question.
  + Feature for showing overall comparison between different taggings of Selection tests data.
  + Showing statistics for change in the tags, that is, deficits and excess, overall.
* **Addon Features**:
  + Generation of questions sets

**1.4 Business Case for the Product**

*Why is this product required? How will it contribute to the goals of your institution? This section can be used when requirements are being negotiated, to assess whether a particular change is a good idea. This section also helps readers understand why certain requirements have been included.*

**To be discussed**

**1.5 Overview of the Requirements Document**

*If your project is small to medium in size, include a summary of the requirements here. This may be a numbered list of the most important requirements. The purpose of this section is to give the reader a general understanding of the requirements and focus attention on the most critical ones. This section may also help point readers to the specific requirements that are of particular interest to them.*

**Please refer to the Scope**

**2. General Description**

This section will give the reader an overview of the project, including why it was conceived, what it will do when complete, and the types of people we expect will use it. We also list constraints that were faced during development and assumptions we made about how we would proceed.

*This section contains a nontechnical description of the project, usually in narrative form, which may serve to acquaint new readers with the purpose of the project. It also sets the stage for the specific requirement listing which follows.*

This project proposes to develop a machine learning based software/web-app for analysis of the data generated during the Online Selection Test for e-Yantra Robotics Competition (eYRC) which is conducted annually by the e-Yantra, Indian Institute of Technology - Bombay.

Data generated during the Online Selection Test contains raw information regarding the performance of the test-takers. This project aims to critically evaluate the raw information and present user readable knowledge set which will be used as performance benchmarking for future Online Selection Tests. This project will revolve around the difficulty taggings of the questions which are used for the Online Selection Test, and one of the core objective of this Project will be to predict the accurate difficulty tagging, which are easy, medium and hard, for each questions, and compare it with the manual tagging provided by the question-masters whom initially designed the questions.

**2.1 Product Perspective**

*Why have you chosen to develop this product? What need does it serve? Who are the primary stakeholders, who is developing the project, and who will benefit from the finished product?*

During a Selection test, it is important to provide questions to the test-takers which are at par in terms of overall difficulty and at the same time the question-master should be well versed to tell the difficulty level of the questions, so that questions with wrongly tagged difficulty levels do not end up to the test-takers.

This project encompasses extensive research in the field of machine learning so as to come up with certain features set which will be used to design a machine learning algorithm to accurately predict the consistent difficulty levels of questions across successive Online Selection Tests. Furthermore the project will employ certain data analytic methods to build a recommender system website which will train the question-master to tag new questions with difficulty with more accuracy. This way personal bias of the question-master won’t influence the manual tagging of the questions.

This project will be developed at individual basis, and will require frequent usability testings to better understand the requirements of a question-master. This will involve my current faculty and may involve my former employer.e -Yantra. I will be requiring the advice of my faculty to come up with better solution for the development of the machine learning algorithm. Due to the nature of non-disclosure agreement signed by the author of this project, few sections of this project will remain restricted to the stakeholder, except for the e-Yantra and the Author.

This project will be a full fledge product, which will combine the Machine Learning area with the Web development area. The product will primarily help my former employer, e-Yantra, to conduct their Selection Test in a more better way, this project will hand them the tools to easily analyse the previously held Selection Tests.

**2.2 Product Functions**

*What does your product do? What activities can users perform while using it? List the main functions that you will build into your product here.*

The product will have two main functions:

1. **Front-end:( intermediate features)**
   1. There will be a Web application where the users can choose which Selection Test data will be processed by the machine learning algorithm for the data analytics generation.
   2. The interface will show the predicted difficulty taggings for the questions which were used in the Selection Tests.
   3. The application will provide the comparative analysis between the manual taggings and the predicted taggings.
   4. The application will provide the comparative analysis between the predicted taggings of different Selection tests.
   5. It will provide questions specific information, such as flagged questions which shows unusually high incorrect attempts in successive Selection Tests, question category-wise accuracy of predicted taggings.
   6. Provide the tool for adding new question with different interfaces to help the users to efficiently tag new questions.
   7. Show visualization for the current ongoing training on the Selection Test data to the user.
   8. Show the changes in the difficulty level of questions, show overall defecit and excess in the question categories caused by the change in the difficulty levels.
   9. User account system with authentication and access restrictions.
   10. Different levels of Users, like moderators, admins, question-masters.
   11. Email system
   12. Team Performance and college performance
2. **Back-end: (Core features)**
   1. Different Machine Learning algorithms will be developed and one with the maximum accuracy will be incorporated with the Web application.
   2. Web-server with database connectivity for deploying the Web application and providing data to the machine learning algorithm for crunching.
   3. Several Restful API links for communication between the machine learning back-end, Web-server and Front-end.

**2.3 User Characteristics**

*Who do you expect to use your finished product, and why? What is their technical background, their training or education, their motivation to use it? What obstacles might they encounter, and what specialized skills will they need?*

This product will be mainly used by my former employer, e-Yantra, and the question-masters who work for them. e-Yantra can use this product as a tool to train the question-master to tag the new question efficiently. The user of this product doesn’t need to be highly technical oriented to use it.

One of the main motivation of this product is to make usability and data analysis a mouse clicks play, provide simple interface with easier to understand the directions. The user will need to know the working of a Online Selection Test. It will be more better if the user knows about the question creation methodology and some important database tables. The user will be required to understand about basic terminology related to the Selection Tests. How the selection test works, etc, difficulty level terms.

The broader target of the product can be those organisation which conducts the online test and will like to have feed-back on the quality of questions which they employed, instead of using the surveys to gauge the views of test-takers on the difficulty of the questions, they can get detailed analysis of the those questions using the responses generated by the test-takers on those questions.

**2.4 General Constraints**

*Did you work under any constraints such as platform or development environment? Did you have to make your product compatible with any existing software or other products currently in use?*

* Development will be done on a workstation using Ubuntu System.
* This product will use the same database which is used by the Selection Test website.
* It will use the test-takers’ details and question details from the Selection Test website.
* This will be designed to work on a Desktop platform, not for any hand-held devices.
* The back-end system will require a fair system configuration for performing the machine learning analysis quickly.
* There will be no public user for this product.

**2.5 Assumptions and Dependencies**

*In this section, list any assumptions you made about your project (for example, did you assume that the finished product would need to be delivered over the internet?). If your project depends on any particular technical infrastructure, or requires administrators or others with specific skills, note that here.*

* There will be certain assumption on the questions which will be later on added in this document, such as where will the new questions go, some database challenges for incorporating previous tags.
* After rigorous testing, the finished product will be handed over to the e-Yantra over the Internet.
* The project will need an administrator who will maintain the website, add new users, etc.
* The administrator will need to understand about the working of the website and the database.

**3. Specific Requirements**

This section of the document lists specific requirements for ***Automated Analysis of Selection Test Data***. Requirements are divided into the following sections:

1. User requirements. These are requirements written from the point of view of end users, usually expressed in narrative form.
2. System requirements. These are detailed specifications describing the functions the system must be capable of doing.
3. Interface requirements. These are requirements about the user interface, which may be expressed as a list, as a narrative, or as images of screen mock-ups.

**3.1 User Requirements**

*List user requirements here.*

* Ability to add new questions,
* Ability to view previous questions and their relevant informations.
* Ability to perform machine learning analysis.
* Ability to export data into a excel document

**3.2 System Requirements***List detailed system requirements here. If your system is large, you may wish to break this into several subsections.*

1. **Servers:**
   1. Python Restful API server (tentative)
   2. Python for running machine learning algorithm
   3. Some Python Libraries, like skLearn etc
   4. MySQL database Server
   5. Apache Server for deploying the Laravel framework based website
2. **Client System:**
   1. Latest Web browser on desktop computer

**3.3 Interface Requirements***List interface requirements here; or include screen mockups. If you use mockups, be sure to explain major features or functions with narrative to avoid confusion or omission of desired features.*

***Will be added in later stages of this project***

**4. Appendices**

*If you wish to append any documents, do so here. You may wish to include some or all of the following:*

* *Personas and scenarios developed for this project*
* *Transcripts of user interviews, observations, or focus groups*
* *Copies of communications which contain user requirements*
* *Original project proposals or other historical documents*
* *Lists of similar projects or products, with notes about how they differ from yours*
* *A list of requirements which were "wish-listed" or marked unfeasible at present*
* *Original screen mockups, if they are relevant*
* Non-disclosure copy
* Original Project Proposal
* Generation of Question Sets after the predicted taggings.

**5. Glossary**

*Include a glossary of definitions, acronyms, and abbreviations that might be unfamiliar to some readers, especially technical terms that may not be understood by end-users or domain-specific terms that might not be familiar to developers.*

* **Difficulty-tags:** 0 for easy, 1 for medium and 2 for hard difficulty. These tags are allotted to every questions.
* **e-Yantra:** it is an initiative by IIT Bombay that aims to create the next generation of embedded systems engineers with a practical outlook to help provide practical solutions to some of the real world problems.
* **eYRC:** Also called e-Yantra Robotics Competition, conduction by e-Yantra.
* **Questions**: In the Selection Test, there are three categories of questions, C programming, Electricals and General Aptitude. Each have 10 questions for each Questions set. each 10 questions are in ratios of 3:4:3 (easy:medium:hard).
* **Question-Master:** user who creates new question for the Selection Test.
* **Question-Set:** Each test-taker gets a set of 30 Questions. Each set will have mutually exclusive questions.

**6. References**

*List references and source documents, if any, in this section.*

***To be added later in stages***

**7. Index**

*If your document is very large, consider compiling an index to help readers find specific items.*

***To be added later in stages***