

**Predictive analytics with online data for WSO2
Machine Learner with the support of Ensemble
method**

Project ID: 16-054

**Software Requirements Specification
(SRS)**

**Author: G.H.G.A. Dias
IT13073060**

**B.Sc. Special (Honors) Degree in Information Technology
Submitted on 01.04.2016**

DECLARATION

I hereby declare that the Software Requirements Specification entitled “Predictive analytics with online data for WSO2 Machine Learner with the support of Ensemble method” submitted to the Sri Lanka Institute of Information Technology is a record of an original work done by me, under the guidance of our supervisor Mr. Lakmal Rupasinghe. This project work is submitted in the partial fulfillment of the requirement for the award of the degree of Bachelor of Science in Information Technology. The results embodied in this report have not been submitted to any other University or Institution for the award of any other degree or diploma. Information derived from the published or unpublished work of others has been acknowledged in the text and a complete list of references is given.

.....

G.H.G.A. Dias
IT13073060

The above candidates have carried out research for the B.Sc. dissertation under my supervision.

Supervisor

.....

Mr. Lakmal Rupasinghe

Table of Contents

1	Introduction.....	6
1.1	Purpose	6
1.2	Scope	6
1.2.1	Objectives	7
1.2.2	Benefits	7
1.3	Definitions, Acronyms, and Abbreviations.....	7
1.4	Overview	8
1.4.1	Goals	9
1.4.2	Users	9
1.4.3	System Overview	10
2	Overall Description.....	10
2.1	Product perspective	11
2.1.1	System interfaces	12
2.1.2	User interfaces	13
2.1.3	Hardware interfaces	16
2.1.4	Software interfaces.....	16
2.1.5	Communication interfaces	16
2.1.6	Memory constraints	16
2.1.7	Operations	16
2.1.8	Site adaptation requirements.....	17
2.2	Product functions.....	17
2.3	User characteristics	22
2.4	Constraints.....	22
2.5	Assumptions and dependencies.....	23
2.6	Apportioning of requirements	23
2.6.1	Essential Requirements	23
3	Specific requirements.....	24
3.1	External interface requirements	24
3.1.1	User interfaces	24
3.1.2	Hardware interfaces	25
3.1.3	Software interfaces.....	25
3.1.4	Communication interfaces	26
3.2	Classes/Objects.....	26

3.3	Performance requirements.....	27
3.4	Design constraints	27
3.5	Software system attributes	27
3.5.1	Reliability.....	27
3.5.2	Availability	27
3.5.3	Security	28
3.5.4	Maintainability	28
3.6	Other requirements	28
4	Supporting information.....	29
4.1	Appendices	29
5	REFERENCES	33

List of Figures

Figure 1 - System overview	10
Figure 2- Home page	13
Figure 3- Visualization of data set	14
Figure 4 - Chart generation interface	14
Figure 5 - Summary statistics interface	15
Figure 6 - Model summary Interface	15
Figure 7- Use case diagram for the ML	21
Figure 8 - Class Diagram	26
Figure 9 - Pseudo code for the selection of an algorithm	29
Figure 10 - Work Breakdown Structure.....	30
Figure 11 - ER diagram for the system	31
Figure 12 - Gantt chart.....	32

List of Tables

Table 1 - Product Perspective	12
Table 2 - Use case scenario to start the WSO2 ML	17
Table 3 - Use case scenario to import data to ML	18
Table 4 - Use case scenario for visually exploring the dataset	18
Table 5 - Use case scenario for training a predictive model	19
Table 6 - Use case scenario for evaluating model	20
Table 7 - Use case scenario for predicting model	20
Table 8- User characteristics.....	22

1 Introduction

This section provides an overview of the SRS document and overall scope description of the research project. Also, the purpose of the document is described and a list of abbreviations and definitions are provided.

1.1 Purpose

This document basically focuses on providing entire requirement set for our proposed system- Predictive analytics with online data for WSO2 Machine Learner. Throughout the document, it describes all the necessary information related to the research project. It also guarantees that the team will develop functionality that has been detailed in the document. The information is organized in such a way that the team members will not only understand the limitations of their work but also what functionality needs to be developed and in what order. It further addresses user interfaces, flow of the project, internal functionalities of software components and various required levels of software system quality attributes. This document is intended for both the stakeholders and the developers of the system. It will be used by developers to implement the functionalities and to ensure traceability of the software, by testers to test the software against the requirements. Also it will be useful for other researches who are interested in implementing this kind of applications.

1.2 Scope

WSO2 Machine Learner is designed for the Enterprise world. It comes as an integrated solution with the rest of the Big Data processing technologies: batch, real-time and interactive analytics. Also, it includes support from data collection, analysis, to communication (e.g. visualizations, APIs, and alerts).

WSO2 ML handles the full predictive analytics lifecycle, including model deployment and management. You can utilize WSO2 Machine Learner to build machine learning models for various tasks, such as fraud detection, anomaly detection, classification etc. It is also quite novice friendly especially for developers, data scientists to quickly implement machine learning algorithms.

1.2.1 Objectives

- Designing architecture for incremental learning and visualizations.
- Creating the incremental learning component.
- Creating interactive visualizations for incremental learning models
- Allowing users to change parameters of the algorithms on the fly (during the incremental learning procedure) by analyzing the models.

1.2.2 Benefits

- User can start with data (in his disk, in HDFS, or in WSO2 DAS)
- Explore the data
- Create a Project and build machine learning models going through a Wizard
- Compare those models and find the best model
- Export that model and use it with WSO2 CEP, WSO2 ESB, or from Java Code.
- Capability of making intelligent decisions
- Sophisticated pattern recognition

1.3 Definitions, Acronyms, and Abbreviations

ESB	Enterprise Service Bus
-----	------------------------

CSV	Comma Separated Value
TSV	Tab Separated Value
JDK	Java Development Kit
RAM	Random Access Memory
GB	Gigabyte
MB	Megabyte
PC	Personal Computer
REST API	Representational State Transfer Application Program Interface
URL	Uniform Resource Locator
ER	Entity Relationship
ML	Machine Learner
SRS	System Requirement Specification
API	Application Program Interface
HDFS	Hadoop Distributed File System
DAS	Data Analytics Server
CEP	Complex Event Processor

1.4 Overview

This SRS document intends to cover all the functional and non-functional requirements of our proposed system. Each of them has been clearly described in detail under 3 main chapters.

The first chapter is a summarization of the general overview of the system. It is not that technical therefore can easily be understood by anyone who reads it. It provides a full description of the project to the users who are interested in predictive analytics, data mining and machine learning. Basically it covers the purpose of the SRS, what the system intends to do, general objectives and goals, benefits, acronyms, abbreviations and overview of the system.

The second chapter describes the overall description in a non-technical way which is understandable by the user. It includes product perspective, product functions, user

characteristics, constraints, assumptions and dependencies and apportioning of requirements. Main target of the product perspective is to find whether the existing system is available in regard for the developing application. Product functions are also described as a summary of all major functions of the application. User characteristics describe the kind of people who will be using the product. Constraints subsection describes the conditions that limit the developer's options. Assumptions and dependencies focus on any assumptions that are made while the apportioning of requirements describes the order in which the requirements are to be implemented when developing the application.

The third chapter of the SRS describes the developer's point of view of the Machine Learner with predictive analytics. It uses technical words and phrases understandable by developers and testers. External interface requirements, performance requirements, design constraints, application attributes and other requirements are also explained in advance.

In conclusion this document will focus on the system deployment and usage procedures in the real world environment and how it will meet the requirements of stakeholders. It can be used as a guide by the development team throughout the development phase.

1.4.1 Goals

- Extract data from CSV or TSV file formats, HDFS or WSO2 DAS
- Use multiple visualizations to explore your data - scatter plots, histograms, Trellis charts, parallel sets and cluster diagrams
- Use feature engineering to pre-process data for better results
- Easy graphical user interface for human-friendly viewing

1.4.2 Users

WSO2 Machine Learner with predictive analytics can be used by any organization or individual who wishes to use it.

1.4.3 System Overview

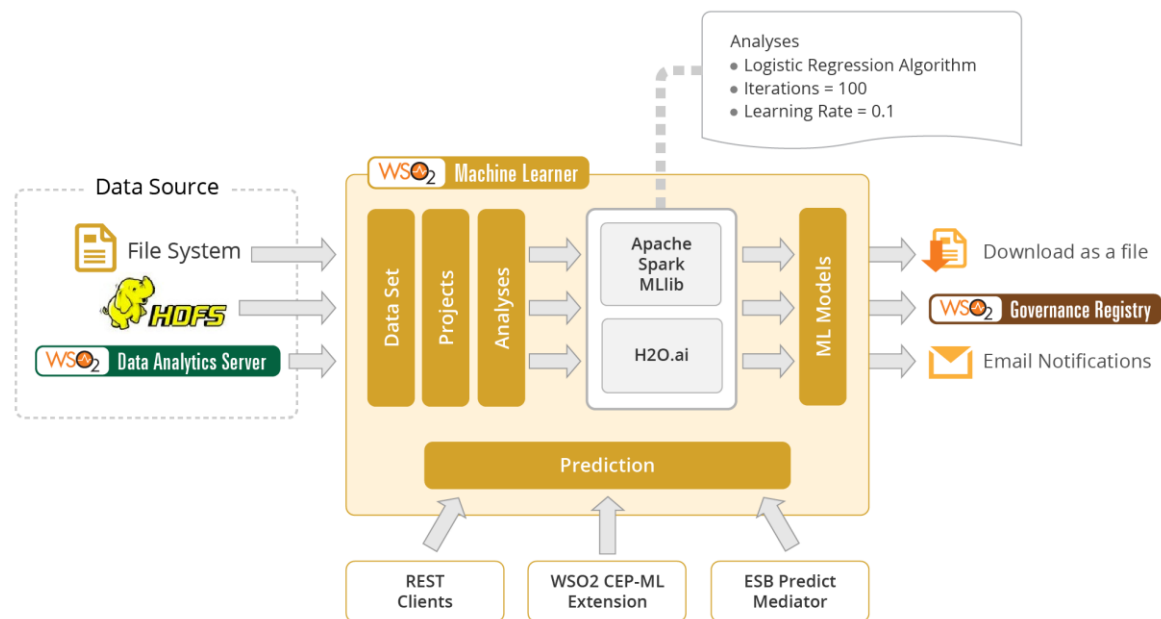


Figure 1 - System overview

2 Overall Description

The generation of data has never been at a higher pace, data which concerns our day to day endeavors such as personal, financial, sales, marketing and so on. Predictive analytics is the use of data, statistical algorithms and machine-learning techniques to identify the likelihood of future outcomes based on historical data. The ultimate goal of predictive analytics is to exceed the descriptive statistics and provide the best assessment on what will happen in the future which in turn will streamline decision making and result in better actions.





In the real world scenarios, many different types of data are acquired sequentially. Traditional way of waiting for data to be collected and identifying patterns is a one-time process. As the new data arrives, these patterns change and the learned patterns need to be evolved accordingly and the decisions based on these models change as well. Concept of machine learning algorithms with streaming data has been emerged under these circumstances. The WSO2 machine learner aims at achieving the following two main objectives; identifying patterns in recent history and updating the patterns with incoming data without catastrophic forgetting. The specialty being this machine learner allows users to change parameters of the algorithm on the fly (during the incremental learning procedure)

WSO2 Machine Learner has a versatile set of characteristics. It can extract features from a dataset made available from a file system, Hadoop Distributed File System (HDFS) or WSO2 Data Analytics Server (DAS). This data is passed on to the Machine Learner Core, which allows you to explore your datasets, pre-process your data and apply various machine learning algorithms to make sense out of it all. Using Apache spark, it then analyzes and builds models with the chosen algorithms.

2.1 Product perspective

WSO2 machine learner will be designed for the enterprise world. It will come as an integrated solution with the rest of the big data processing technologies: batch, real-time and interactive analytics. Also it will include support from data collection, analysis to communication (visualizations, APIs and alerts). Hence it is a part of a complete analytical solution. WSO2 Machine Learner will fundamentally address predictive analytics lifecycle including model deployment and management while also catering to changing parameters of the algorithm on the fly. Some of the special features are:

- Support for Deep Learning and Neural Networks
- Support for out of the Box Anomaly detection using Markov Chains and Clustering
- Support to data cleanup and preprocessing using Data Wrangler and SparkSQL
- Support for out of the box ensembles that let you combine models
- Improvements to pipeline to warn the user on cases like class imbalances in classifications

Features	WEKA	ORANGE System	Existing WSO2 ML	Proposed WSO2 ML
User Friendly Interfaces				

Outlier detection when generating graphs	X	X	X	✓
Model Generation	X	X	✓	✓
Can configure with spark	X	X	X	✓
Display prediction result rather than value	X	X	X	✓
Validate File Formats	X	X	X	✓
Improved Performance	X	X	X	✓
Successfully displaying cluster diagrams	X	X	X	✓
Use Streaming data	X	X	X	✓
Use ensemble method	X	X	X	✓

Table 1 - Product Perspective

2.1.1 System interfaces

- Streaming Data Extraction Interface
- View data interface

- Data Transformation Interface
- Evaluating model and predicting interface
- Data visualization interface

2.1.2 User interfaces

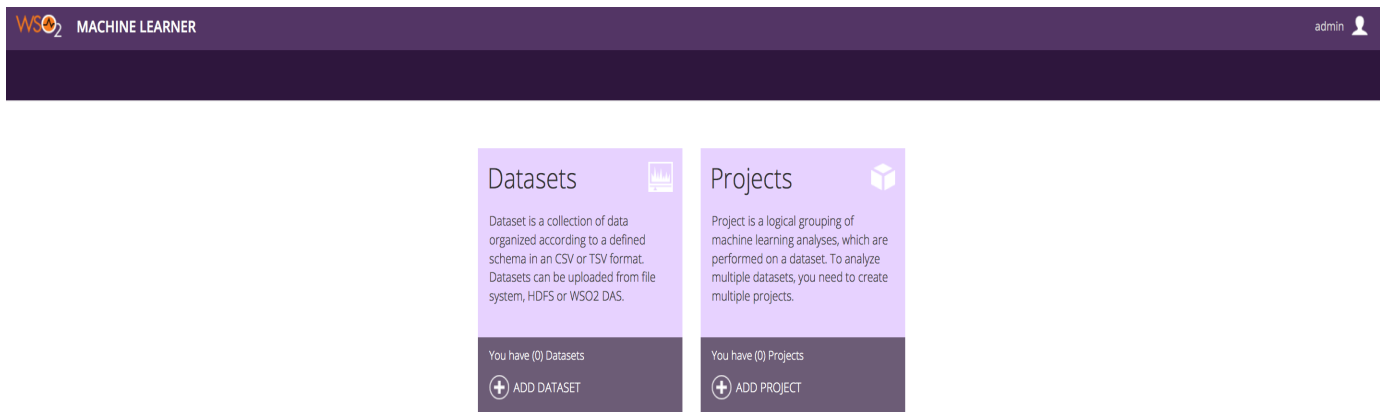


Figure 2- Home page

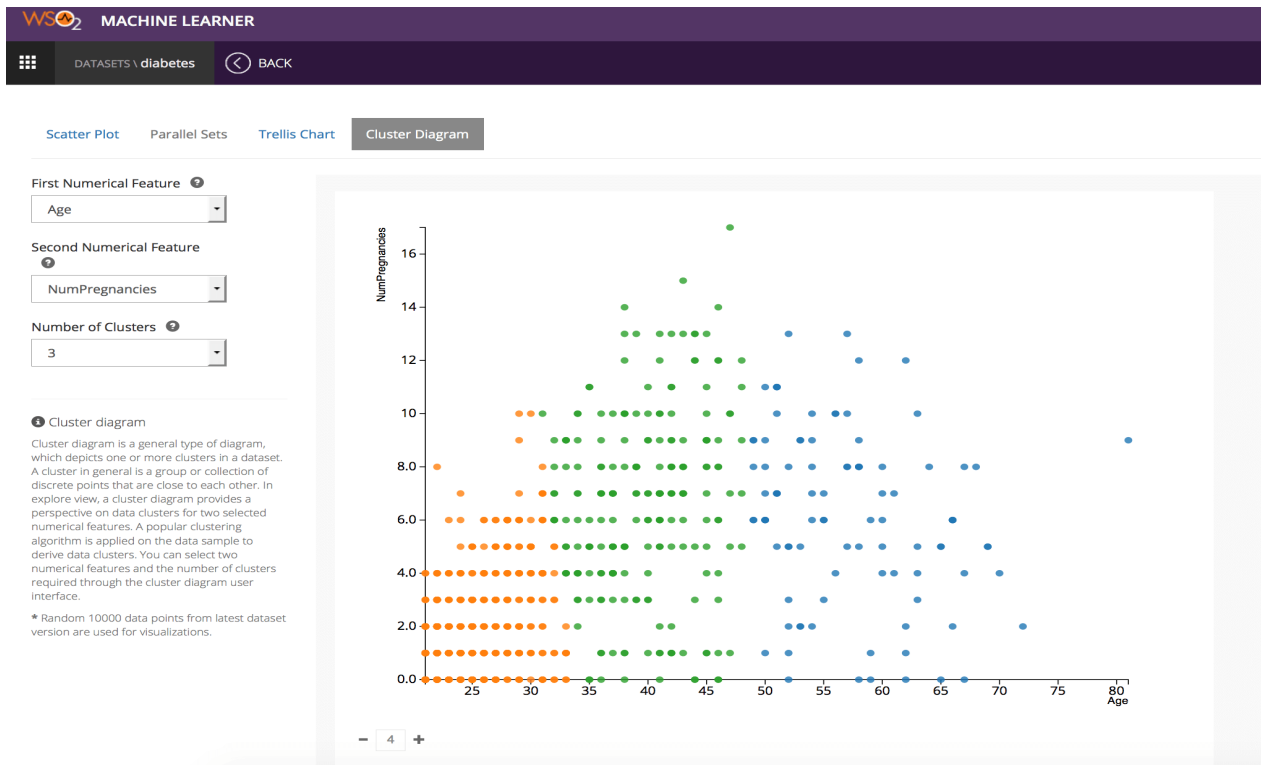


Figure 3- Visualization of data set

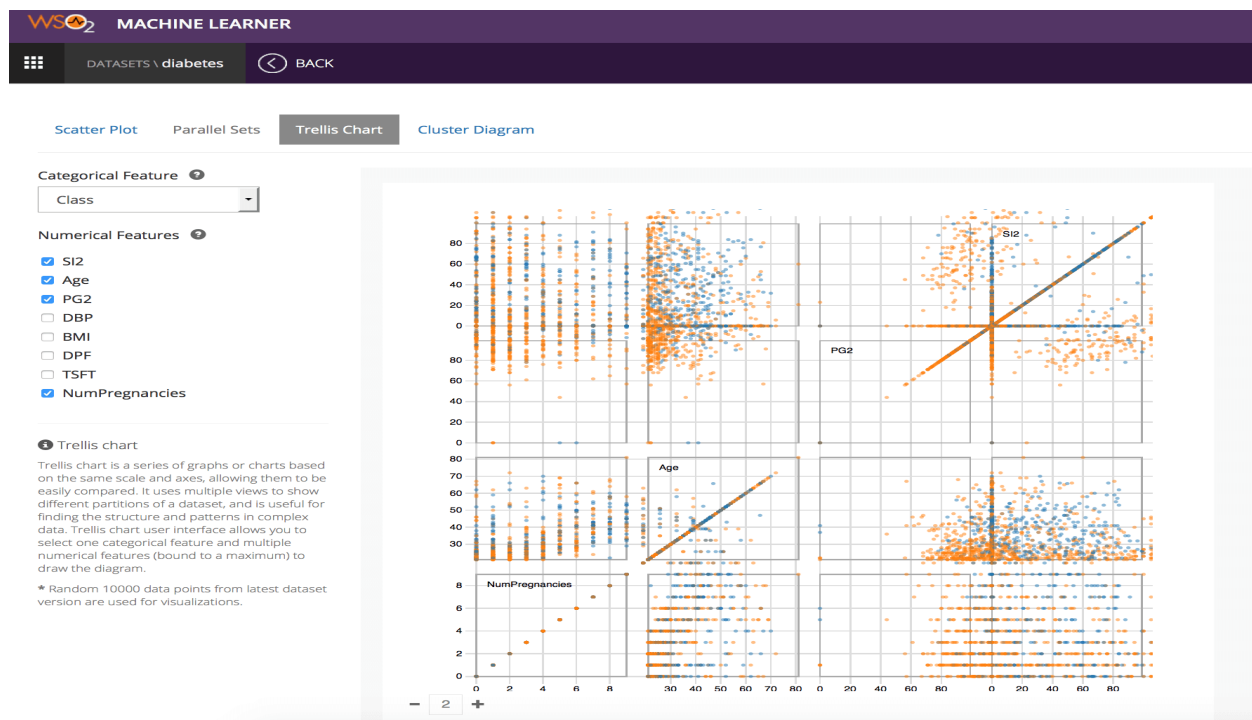


Figure 4 - Chart generation interface

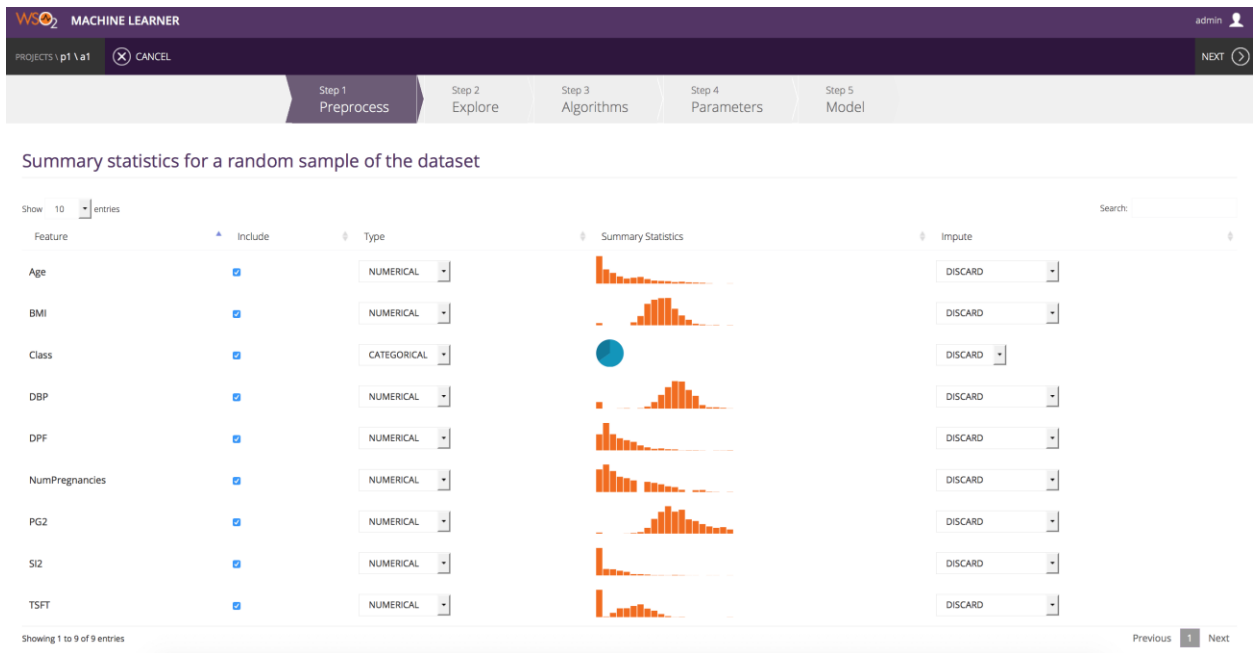


Figure 5 - Summary statistics interface

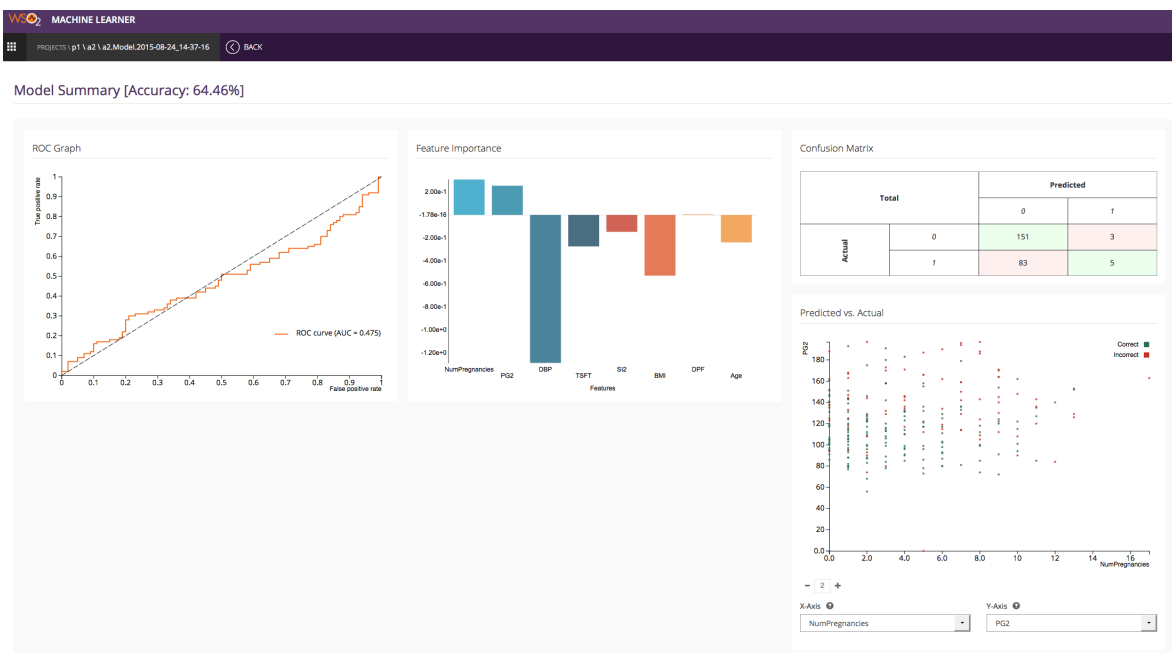


Figure 6 - Model summary Interface

2.1.3 Hardware interfaces

No special hardware interfaces are used for the system.

2.1.4 Software interfaces

- Oracle Java SE Development Kit (JDK) 1.6.24 or later / 1.7.*
- Apache ActiveMQ JMS Provider 5.5.0 or later
- Apache Ant 1.7.0 or later
- SVN Client
- Apache Maven 3.0 or later
- JavaScript enabled Web Browser
- MySQL

2.1.5 Communication interfaces

- Internet
- Database connection interface

2.1.6 Memory constraints

- RAM of 2GB or higher

2.1.7 Operations

- Download WSO2 ML from WSO2 website
- Make sure you have java 7 installed in your machine and set JAVA_HOME
- Unzip the pack and run bin/wso2server.sh from the unpacked directory. Wait for ML to start.
- Go to <https://hostname:9443/ml> and Login using username admin and password admin.
- Now you can upload your own dataset and follow along with the wizard. You can find more info from the User Guide. However, Wizard should be self-explanatory.

2.1.8 Site adaptation requirements

- The user machine must have Oracle JDK 1.7 or a newer version installed on it
Special note: OpenJDK is not recommended
- To build the product from the source distribution Apache maven needs to be installed.

2.2 Product functions

Use case Name	Start the WSO2 ML
Pre –Condition	An Oracle 7/8 compliant platform A modern JavaScript enabled web browser
Actor	User
Main success scenario	1. Download the WSO2 ML pack and unzip it 2. Move to the bin directory and execute the WSO2 server script 3. Once the ML server has started locate the ML console URL and open it from a web browser 4. Login with the credentials 5. You are now directed to the ML home page
Extensions	4a. Invalid login credentials

Table 2 - Use case scenario to start the WSO2 ML

Use case Name	Import data to ML
Pre –Condition	ML should be properly installed in the user’s machine
Actor	User
	1. Click on “Add dataset” in the home page to import data
Main Success Scenarios	2. Fill the necessary fields 3. Click “Choose file” and select the data file 4. Once all the fields are filled click on “Create dataset”
Extension	2a. The fields aren’t filled properly 3a. The file type can only be CSV or TSV

Table 3 - Use case scenario to import data to ML

Use case Name	Visually explore the dataset
Pre –Condition	Dataset should be imported to the ML
Actor	User
Main Success Scenarios	1. Click on “Explore dataset” 2. Choose the type of chart you want; Scatter plot, Trellis chart or Cluster diagram
Extension	2a. The type of chart isn’t specified properly

Table 4 - Use case scenario for visually exploring the dataset

Use case Name	Training a predictive model
Pre –Condition	Dataset should be imported to the ML
Actor	User
Main Success Scenarios	<ol style="list-style-type: none"> 1. Click on “Add project in the home page” 2. Fill the necessary fields and choose the data set you have already provided from the dropdown 3. Click “Create Project” 4. Provide a name for the analysis and click on “Create analysis” 5. Change the values as you wish and click Next 6. You can again visually explore the dataset if you want, else click Next 7. Select the algorithm to be used when training the model and fill the other values accordingly and click Next 8. Select the version of the dataset and Proceed. 9. Click on Run to build the models 10. Once the model building process is complete the status will be displayed as Complete
Extension	7a. The algorithm details are not specified properly

Table 5 - Use case scenario for training a predictive model

Use case Name	Evaluating models
Pre –Condition	Predictive model should be generated
Actor	User
Main Success Scenarios	<ol style="list-style-type: none"> 1. From the home page click on “Projects” 2. Under the projects click on “Models” 3. Under the models click on “View” 4. User will be directed to the prediction results with the accuracy

Table 6 - Use case scenario for evaluating model

Use case Name	Predicting models
Pre –Condition	-
Actor	User
Main Success Scenarios	<ol style="list-style-type: none"> 1. Navigate to the page where the models are listed 2. Click on “Predict” 3. Fill the values with necessary data 4. Obtain your prediction

Table 7 - Use case scenario for predicting model

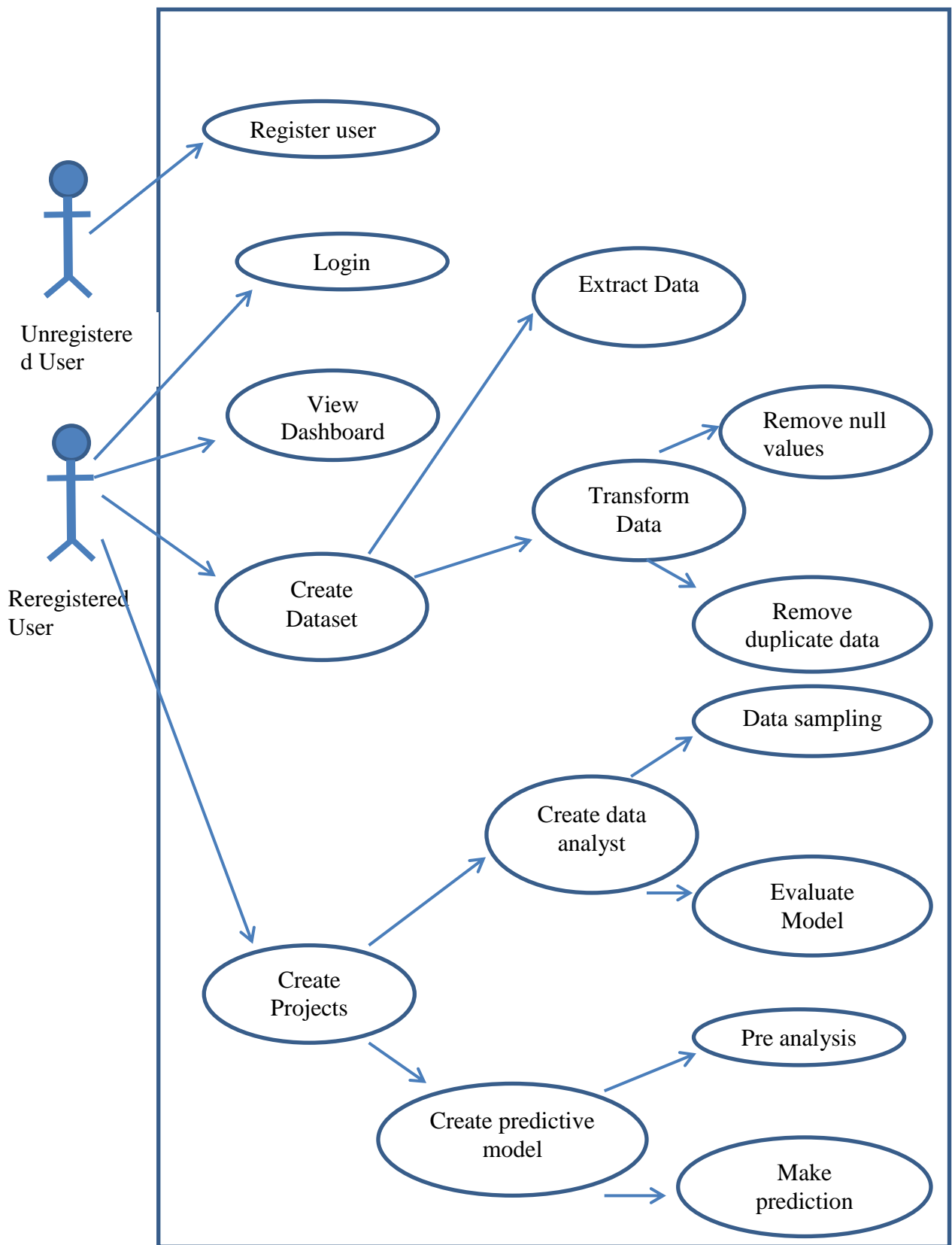


Figure 7- Use case diagram for the ML

2.3 User characteristics

There is no particular user for which the Machine Learner is intended. It is useful for any individual or organization who is interested in applying predictive analytics with machine learning for their any of their purposes. Especially enterprises and other businesses could use this to gain benefits for their respective organizations financial and sales wise.

Specific users include: data scientists, researchers, developers, statisticians and so on.

User type	Description
Data scientist	Person who wishes to study years and years of collected data and identify patterns
Researcher	Person who is studying about a particular field could study collections of old data and come up with solutions
Statistician	Statisticians are concerned with the collection, analysis and interpretation of quantitative data
Developer	Predictive analytics is useful for developers

Table 8- User characteristics

2.4 Constraints

- A major constraint that the development team has to face is the lack of time. The time allocated for developing this project is only 10 months during which time they have to learn other subjects as well.
- Lack of resource personnel
- All the tools and technologies used for the development should be open source.

2.5 Assumptions and dependencies

- Users who are using this have sufficient knowledge about computers and internet.
- Users have some idea about predictive analytics and machine learning.
- The machine learner is successfully installed in the user's PC.
- There is an active internet connection.
- There is sufficient memory and processing power in all user PC's.

2.6 Apportioning of requirements

The most important requirements which are addressed in the section 3 are to be implemented in the system in its first release. This also contains the services and facilities for the user to satisfy the following requirements.

2.6.1 Essential Requirements

- Data exploration: Ability to convert datasets into a format, which is suitable for machine learning algorithms (data preprocessing)
- Model generation: Support various machine learning algorithms in classification, numerical prediction and clustering spaces.
- Model comparison: Figures that help you to compare models across a machine learning project
- Prediction: Generated models can be used to perform predictions using a REST API client, WSO2 CEP extension or WSO2 ESB mediator.

3 Specific requirements

3.1 External interface requirements

3.1.1 User interfaces

- **Streaming Data Extraction Interface:**

This is the first stage of the data acquisition process. The relevant streaming data should be loaded into the database which has collected using APIs. The source file should be in from a permitted file type. Then name the dataset and name the version of the dataset and finally, click create dataset to start the data extraction process.

- **View Data Interface:**

In this interface user can see the actual data in a more attractive way. Collected streaming data will be represented in Scatter plot, Trellis Chart and Cluster diagram. User can either transform the displayed data or analyses them from there.

- **Data Transformation Interface:**

This is where the data transformation is done. First the user must select the name of the field which should be transformed. Then the form of transformation needs to be specified. Transformation can be in different forms like correcting data that is incorrect, out-of-date, redundant, incomplete, or formatted incorrectly. Finally user has to select the date range which the data should be.

- **Evaluating model and predicting Interface:**

This interface is used to apply the algorithm to the dataset which has transformed earlier. User can select the k-means clustering algorithm to apply the evaluation to the previously created analysis. When create the analysis machine will get trained according that that dataset. User can change hyper parameters according to k- means clustering algorithm and generate samples from a whole population data. Finally using the predict interface user can select the project where they were working on (which has created the dataset), select the model and then predict the value.

- **Data Visualization Interface**

Finally the predictive value will be represented in Scatter plot, Trellis Chart and Cluster diagram. User can make decisions by exploring these diagrams.

3.1.2 Hardware interfaces

No special hardware interfaces are used for the system.

3.1.3 Software interfaces

- Oracle Java SE Development Kit (JDK) 1.6.24 or later / 1.7.*
To launch the product as each product is a Java application
- Apache ActiveMQ JMS Provider 5.5.0 or later
To enable the product's JMS transport and try out JMS samples. The ActiveMQ client libraries must be installed in the product's classpath before you can enable the JMS transport.
- Apache Ant 1.7.0 or later
To compile and run the product samples
- SVN Client
To check out the code to build the product from the source distribution. If you are installing by downloading and extracting the binary distribution instead of building from the source code, you do not need to install SVN.
- Apache Maven 3.0 or later
To build the product from the source distribution (both JDK and Apache Maven are required). If you are installing by downloading and extracting the binary distribution instead of building from the source code, you do not need to install Maven.
- JavaScript enabled Web Browser
To access each product's Management Console. The Web Browser must be JavaScript enabled to take full advantage of the Management console.

- MySQL

My-SQL is used as the database management system. As the project is basically based on data mining extraction and transformation, My-SQL would be used regularly for major operations of the system.

3.1.4 Communication interfaces

- Internet: We need Internet here because we are going to get streaming data (online data) to feed the Machine Learner

3.2 Classes/Objects

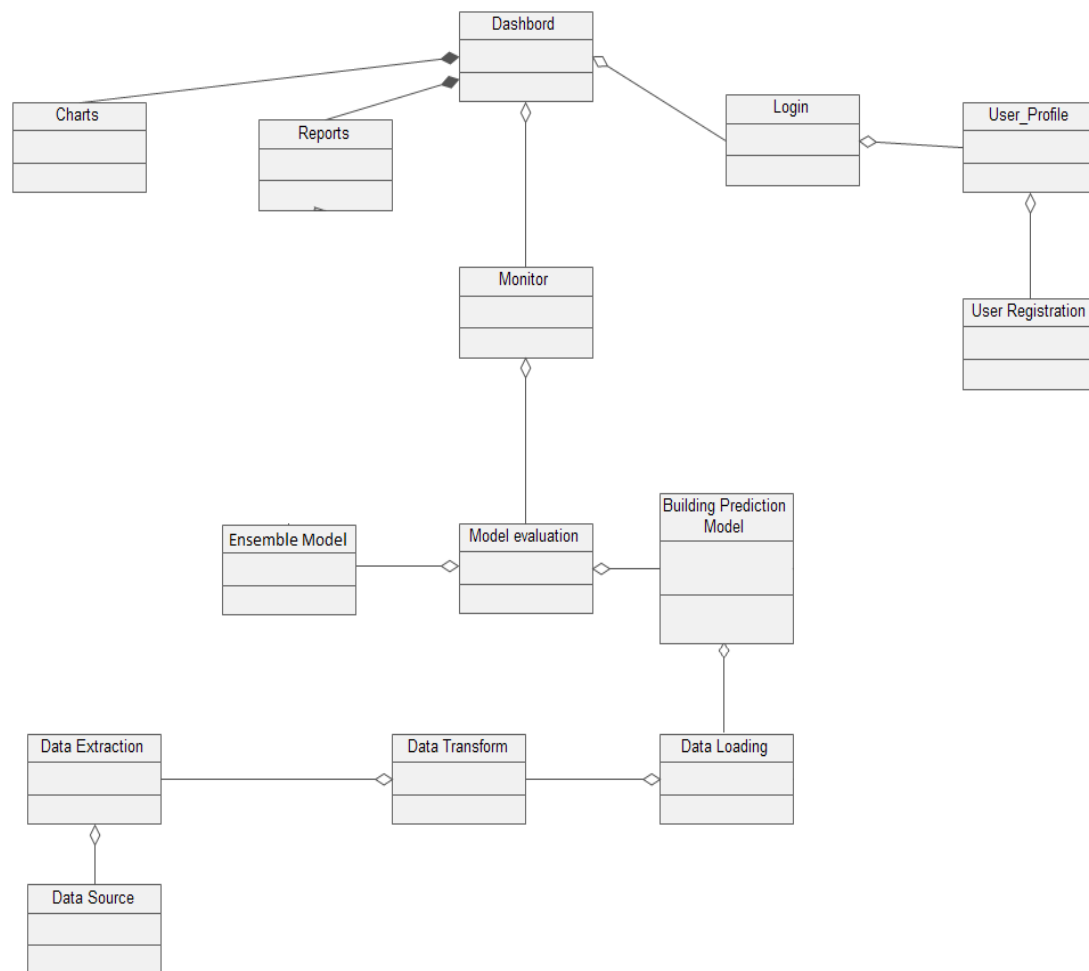


Figure 8 - Class Diagram

3.3 Performance requirements

It is expected that the proposed system will perform all the requirements stated under the functional requirements section. Some performance requirements identified are listed below:

- Predictive list should be generated within an hour.
- Based on the derived predictive lists, reports must be generated within 10 seconds.

3.4 Design constraints

- A Major constraint that we will be facing is the limitation of available time. The project group is expected to complete the project within 10 months.
- Predictive model building, evaluation, testing and all other operations will be carried out using some old datasets.

3.5 Software system attributes

3.5.1 Reliability

Reliability is the ability of the system to run with minimum number of failures. The system has to go through thorough testing i.e. application must be tested by fixing each and every possible bug. Each and every component of the system will be tested and finally integrated system also will be tested to make sure the desired output is obtained. System output also has to be tested to make sure the output is meaningful. Since this is a tool for predictive analytics, the reliability of the system is much anticipated.

3.5.2 Availability

WSO2 ML with predictive analytics can be accessed at any time by anyone who wishes to use it. All the necessary information can be viewed by the users and there is no time restriction imposed on the system. It is open source and free to download.

3.5.3 Security

Security is a crucial non-functional requirement of the proposed system. System is expected to keep all critical information in a very secure manner. All critical data will be secured during storage and transmission through proper access controls. High security will be provided by authenticating and authorizing users.

3.5.4 Maintainability

High maintainability is one of the key virtues of stable and standard products. Even in product implementation of the proposed system, we are focused on creating a highly maintainable system. The standards of the coding practices will be followed throughout system implementation and it will minimize the bugs as much as possible. Whenever there are new requirements, the system can be modified to accommodate these requirements by maintaining the stability of the system.

3.6 Other requirements

- Data acquisition:

The data for the predictive model building, evaluation and testing should be acquired from valid sources, otherwise it can affect the accuracy of the model.

- Use of open source technologies

The product will be developed only using open source technologies.

4 Supporting information

4.1 Appendices

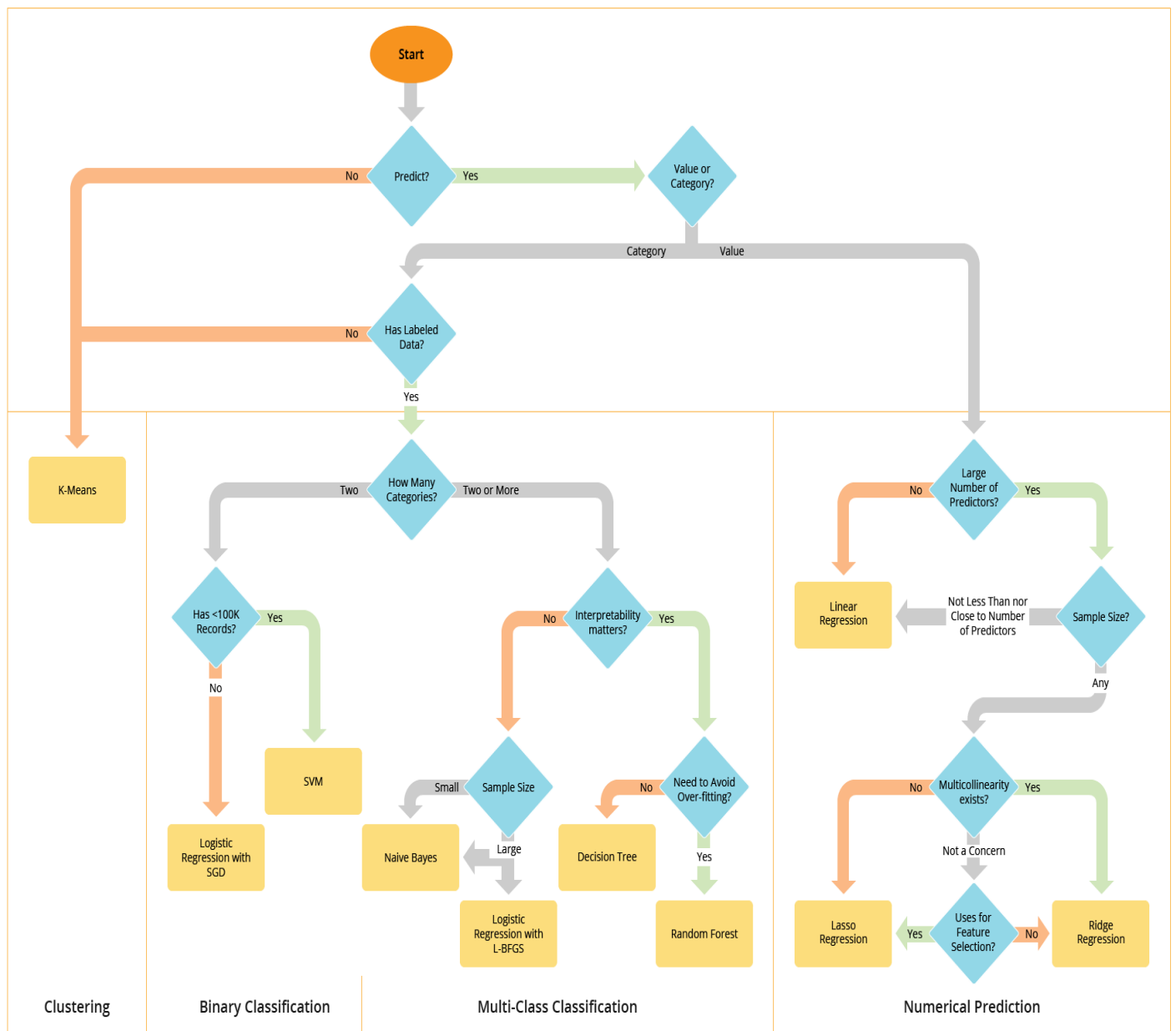


Figure 9 - Pseudo code for the selection of an algorithm

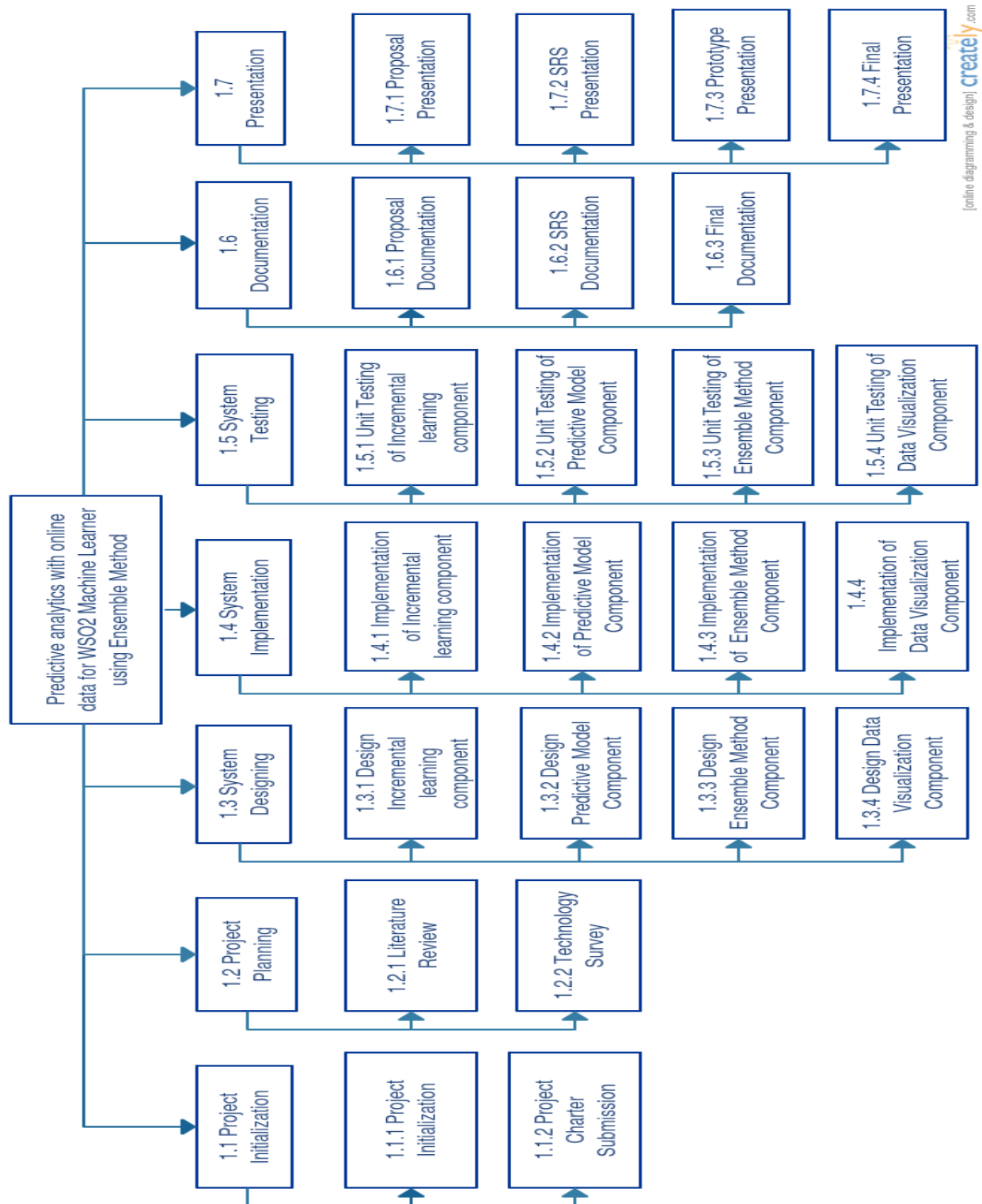


Figure 10 - Work Breakdown Structure

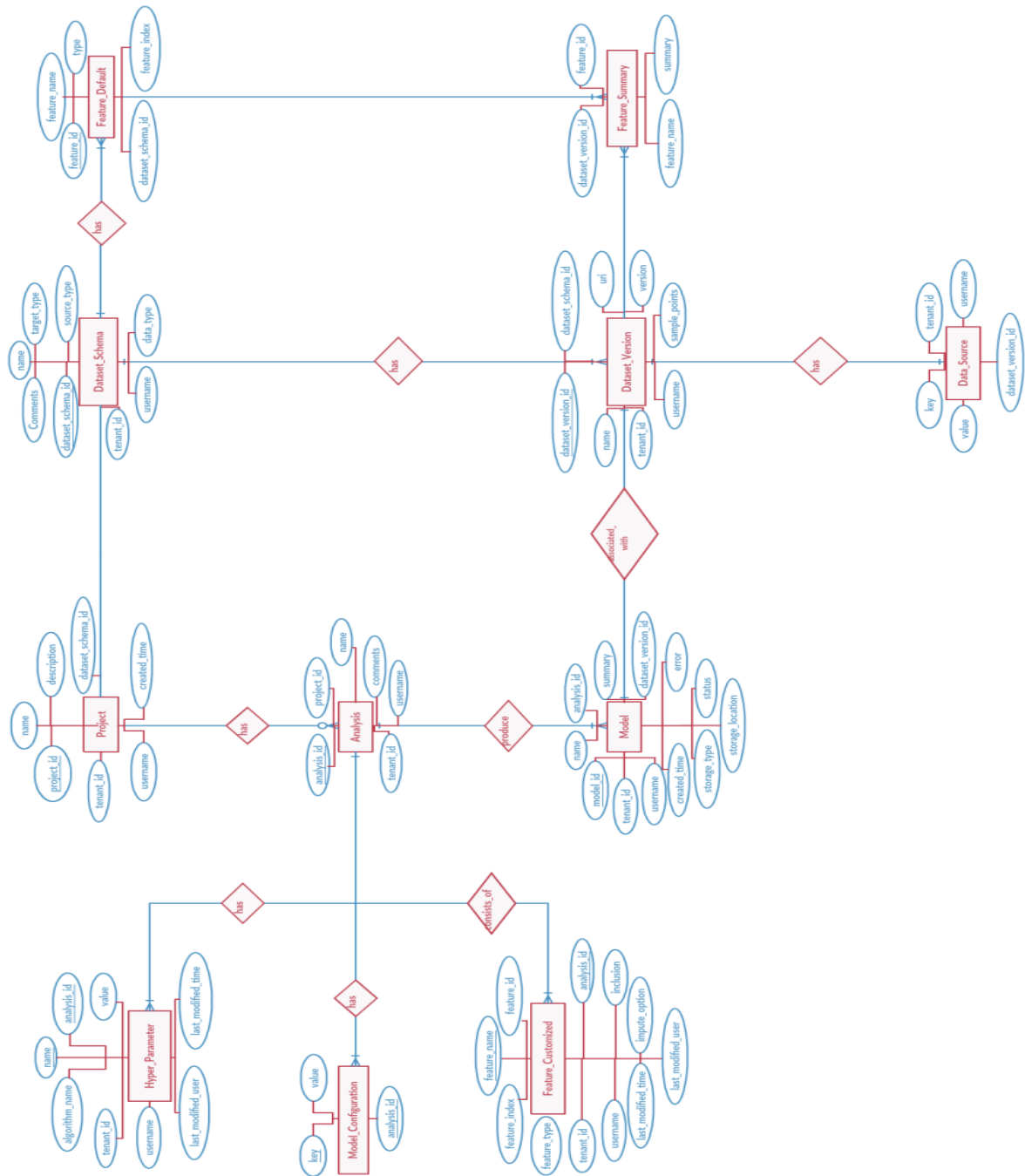


Figure 11 - ER diagram for the system

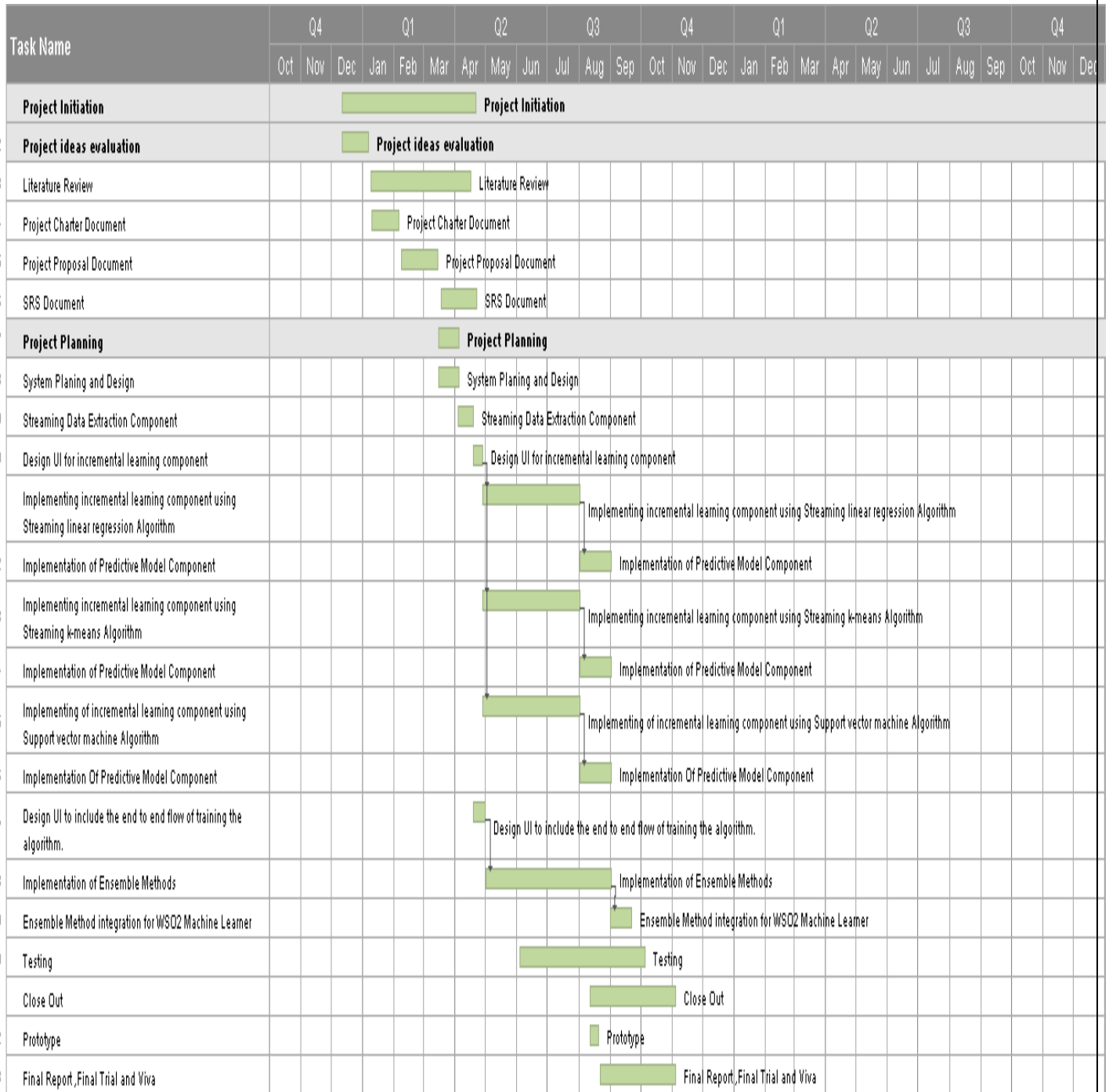


Figure 12 - Gantt chart

5 REFERENCES

- [1] [Online] Available: <http://whatis.techtarget.com/definition/machine-learning>
[Accessed: March 8, 2016]
- [2] [Online] Available:
[https://docs.wso2.com/display/GSoC/Project+Proposals+for+2016#ProjectProposalsfor2016-Proposal6:\[ML\]PredictiveanalyticswithonlinedataforWSO2MachineLearner](https://docs.wso2.com/display/GSoC/Project+Proposals+for+2016#ProjectProposalsfor2016-Proposal6:[ML]PredictiveanalyticswithonlinedataforWSO2MachineLearner)
[Accessed: February 29, 2016]
- [3] [Online]. Available: <https://docs.wso2.com/display/ML110/Features> [Accessed: February 29, 2016]
- [4] [Online]. Available: The Wall Street Journal, Thomas H. Davenport [Accessed: March 1, 2016]
- [5] [Online]. Available: <http://wso2.com/blogs/thesource/2015/11/have-you-checked-out-the-wso2-machine-learner-yet/> [Accessed: March 16, 2016]
- [6] [Online]. Available: <http://nirmalfdo.blogspot.com/2015/07/sneak-peek-into-wso2-machine-learner-10.html> [Accessed: March 16, 2016]
- [7] [Online]. Available: <https://iwringer.wordpress.com/2015/09/25/wso2-machine-learner-why-would-you-care/> [Accessed: March 20, 2016]
- [8] [Online]. Available: <https://docs.wso2.com/display/ML110/Features> [Accessed: March 28, 2016]
- [9] [Online]. Available: <http://wso2.com/products/machine-learner/> [Accessed: March 25, 2016]
- [10] [Online]. Available: <http://dev.datasift.com/docs/api/streaming-api> [Accessed: March 20, 2016]