

Review Document – Requirements Document

Group 03

Paul Fink Jens Henninger Florian Jennewein
Daniel Maier

November 26, 2015

1 Introduction

This is our review of Group 03.

2 Miscellaneous

2.1

The glossary entries are precisely.

2.2

The UR/SR are well, clearly and completely formulated; except the negative mentioned ones.

2.3

The scenarios are well described.

2.4

The structure of the whole document is very well.

2.5

A kind of documentation/help for learning administration and general use is mentioned in UR/SR(a point weve not mentioned).

2.6

Following entries are missing:

- weka, smo,

3 Problems in the requirements

3.1 General observations

3.1.1

User Requirements are not linked to non-functional system requirements. Also some functional requirement are missing their link to a user requirement (For Example FR002)

3.1.2

The difference between priorities is not explained. If there were only two, A and B for example it would be clear that priority A is a must have and priority B is a nice to have. But since there are three different priorities the distinction between those three is not clear.

3.2 Requirement specific observations

3.2.1

UR002: *There shall be no monetarization of the software, such as advertisements.* (corresponding system requirement: NFR034)

This Statement does not affect the development of the application. Furthermore UR001 already states that the finished product will be published as an open-source product, implication that there will be no monetization.

3.2.2

UR003: *The language of the website shall be English.*

Although this is true, the client wants the whole application to be in english. So not only the website, but also the installation process etc. shall be in English. In the corresponding system requirement this mistake is fixed:

NFR007: *The system language is English.*

3.2.3

UR010: *The website should offer a version for mobile devices.*

As far as we understood it, the website should be readable via mobile devices. There does not need to be an extra version for them. In general it seems improbable that users would want to do data mining with a mobile devices since they have very limit disk space for trainings and/or test data.

3.2.4

UR011: *Statement The system should be accessible through a command line interface via SPARQL.*

The database should be accessible via SPARQL not the whole application. Same

goes for the corresponding system requirement:

NFR009: *The system should be accessible through a command line interface via SPARQL.*

3.2.5

UR013: *The system shall run on desktop environments. Already existing server infrastructure provides 32GB of RAM and Octa-Core CPUs.*

This requirement is somewhat poorly stated: The server client will run on servers with at least 32GB of Ram and Octa-Core CPUs.

3.2.6

Multiply times statements listed of things that will not be done. For example:

UR020: *The system infrastructure does not need to be redundant.* (Other statements with the same Problem: UR014, UR015, UR018, UR024(last part), UR029, UR036, NFR011, NFR027, NFR033)

3.2.7

UR021: *Data sets, models and result sets shall be organized in packages which are each assigned a unique ID.*

The last part about assigning unique Ids is already a design decision and should therefore not be included.

3.2.8

UR022: *Package contents shall be accessible through their ID.*

Same here as above. The Statement should probably be more like this: Users shall be access each package respectively

3.2.9

UR023: *Different installations of the system shall be able to exchange models.*

Although this is true, the requirement is suppose to be more comprehensive. Not only models, but also test data shall be exchangeable.

3.2.10

UR028: *Settings chosen by a user shall be saved, independently for each user.*

How else would you save it but independently?

3.2.11

UR034: *The software shall save queries and results per user, so they can be accessed in the future.*

What kind of queries are mentioned here? Saving all queries to the db is surely not wanted.

3.2.12

UR039: *Users shall be notified when the systems storage or memory are full.*

What is the difference between systems storage and memory? If there is none why are both mentioned?

3.2.13

UR043: *The terms and conditions shall be changeable by the administrator.*
(Corresponding system requirement NFR039)

What terms and conditions. The terms of the registered users are not restrictable. For example their algorithm may always run indefinitely.

3.2.14

UR016, UR031, UR046

These three requirements are all about adding algorithms to the application and have somehow different priorities. It seems like there was a misunderstanding, that plugins and uploaded algorithms are not the same thing, but they are. They are also not a must have but a great addition to the application. We gave it therefore priority B.

3.2.15

UR047: *The implemented algorithms offered to the user shall be extensible and modifiable.*

As far as we understood it, the exact opposite is the case. Already implemented algorithms shall be unchangeable.

3.2.16

UR053: *The algorithms shall be executed wholly on the server.*

Where else would they be executed? A somewhat unnecessary statement.

3.2.17

UR057: *Data sets shall be partitionable after upload.*

Is this the same statement as UR044? If though it is redundant. Otherwise partitionable how and why?

3.2.18

UR061: *The raw data sets shall be downloadable.*

Redundant statement, already mentioned in UR037.

3.2.19

NFR012: *The implemented algorithm shall include mainly classification algorithms.*

What is mainly in this case? System requirements should be specific, therefore using a statement like: 80% of the implemented algorithms should be classification algorithms would be better. But in general this seems like a rather strange requirement, since the application will have exactly three algorithms (SMO, Random forest and J48), when it comes out and all additional algorithm must be approved by an administrator. So this seems to be a requirement for the administration and not the developer.

3.2.20

NFR013: *The implemented algorithms shall also include clustering algorithms.* Same as above.

3.2.21

NFR014: *The minimal system requirements demand a desktop environment or server with following specifications: i5 CPU with 8GB Ram, 500 GB storage, and a GNU/Linux operating system.*

And NFR015: *The recommended system requirements for the server are as follows: Intel Xeon CPU with 32GB Ram, 2 TB storage, and a GNU/Linux operating system.*

These two requirements seem to contradict each other, unless NFR014 is not meant for the server client, but for the use of the application, which would be rather strange, since the application should only require a common browser. Also mobile device tend to have hard drives with less 500GB. 8GB Ram is also a rarity.

3.2.22

NFR016: *At most, two algorithms shall be executed at a time. All further requests by users must be enqueued.*

Two algorithms per user or per server? As far as we understood it the administration may choose how many algorithms can run simultaneously on the server.

3.2.23

NFR017: *Once a day garbage collection is triggerd to search expired guest accounts, session recordings, and shared files.*

Guess in this case are anonymous users? If so there is only one account (as stated in NFR031: Unregistered users share an anonymous user account.). Otherwise you have added a fourth user group, which was not required.

3.2.24

NFR021: *The system shall be available 24/7.*

Directly contradicts NFR019, NFR020, NFR022, NFR023, NFR024. There is also now a problem with the priorities. Since NFR021 is a way stronger statement then the other four and therefore why harder to realize but somehow it has Priority A whereas NFR022-NFR024 only have priority B.

3.2.25

NFR037: *The system shall be expendable after deployment.*

First of all I am guessing it is suppose to say expandable. Sorry but this is a rather funny typo. This statement is greatly open for interpretation. The requirement should probably specify what parts of the system are expandable.

3.2.26

NR011: *The user menu allows the user to set read and write rights for data sets, models, results or whole packages owned by them to user groups.*

As far as we understood it the user may only change permission per package.

3.2.27

FR017: *The system must give the opportunity of user-registration via e-mail-address, username, and password.*

Redundant statement, already mentioned in FR004 (thou this statement has more detail)

3.2.28

F018: *The system supports registered and unregistered users.*

Redundant to FR004+FR005

3.2.29

FR019: *Registered shall be able to store their settings made at previous logins. Unregistered users always start with default settings and changes they made are only valid for their current session.*

Since there is only one anonymous account its settings must always be fix. If there are two anonymous user the system cannot differentiate between seem so if one would make change the other one would be affected by them, which is not desired.

3.2.30

Scenario 3.4: Missing what can go wrong for the unregistered user. For example his upload data may exceed the limit given by the administration and therefore the system returns an error.

4 Missing requirements/scenarios:

4.1

There are no UR/SR concerning constraints of the maximum upload file size.

4.2

There are no UR/SR concerning constraints of the allowed maximum number of simultaneously running algorithms per user.

4.3

There are no scenarios for system administration.

4.4

There are no NFR concerning WEKA or RDF.

4.5

There are no Ethical/Legislative NFR.

4.6

There are no UR/SR concerning userfriendly and graphical solutions. Solution shall be presented in usual formats.

4.7

5 Inconsistencies

5.1

The majority of the Environmental NFR are FR.

5.2

FR 1-3 are NFR.

5.3

There is no connection between NFR and UR.

5.4

There is no structure in the order of FR.