Requirements document

SQUID

Helsinki 26th February 2005

Software Engineering Project

UNIVERSITY OF HELSINKI
Department of Computer Science

Course

581260 Software Engineering Project (6 cr)

Project Group

Mikko Jormalainen
Samuli Kaipiainen
Aki Korpua
Esko Luontola
Aki Sysmäläinen

Client

Lauri J. Pesonen

Project Masters

Juha Taina
Jenni Valorinta

Homepage

`http://www.cs.helsinki.fi/group/squid/`

Change Log

Version	Date	Modifications
0.1	9.2.2005	First version (Aki Sysmäläinen)
0.2	18.2.2005	Use case list (Samuli Kaipiainen)
0.3	21.2.2005	Some use cases expanded (Samuli Kaipiainen)
0.4	22.2.2005	User requirements (Mikko Jormalainen)
0.5	23.2.2005	More use cases expanded (Samuli Kaipiainen)
0.6	25.2.2005	All use cases expanded (Samuli Kaipiainen)
0.?	??.2.2005	More user requirements, architecture (Esko Luontola) ...
0.8	26.2.2005	Use case diagram (Samuli Kaipiainen)

Contents

1 Introduction **1**

1.1 Glossary . . . 1

2 Overview **1**

3 Use cases **1**

3.1 Measuring . . . 2

3.2 Exporting . . . 5

3.3 Printing . . . 5

3.4 Projects . . . 6

3.5 Sequences . . . 7

4 User requirements **9**

4.1 Functional requirements . . . 9

4.1.1 Magnetometer . . . 9

4.1.2 Project files . . . 10

4.1.3 Measurements . . . 11

4.1.4 Others . . . 12

4.2 Quality requirements . . . 12

4.3 Environment . . . 13

4.4 Maintainability . . . 13

4.5 Restrictions . . . 13

5 System requirements / Functions **13**

5.1 System restrictions . . . 14

6 User interface **14**

6.1 Goal-derived use cases . . . 14

6.1.1 Goal-derived use case 1: Erkki... . . . 14

6.1.2 Goal-derived use case 2: Fabio... . . . 14

6.1.3 Goal-derived use case 3: Tomas... . . . 14

7 Architecture overview **14**

8 External interfaces **15**
8.1 Existing program 15
8.2 Hardware control protocols 15

9 Validation **16**

Appendices

1 Automated Sample Handler System Protocol

2 Automatic Sample Degaussing System Protocol

3 Superconducting Rock Magnetometer Protocol

1 Introduction

This document describes client requirements and system requirements for a SQUID magnetometer program that will be designed and implemeted as a software engineering student project at University of Helsinki at the Computer Science Department. The client is the Department of Geophysics.

This document serves as a contract between client and us..

Expected readership of this document here..

1.1 Glossary

Technical terms here..

2 Overview

A brief overview of the problem domain..

3 Use cases

Describes planned use cases for the program. Use cases are derived from user interface prototype and user requiremets. All use cases are perfomed by any ordinary user and in program main screen. A simplified use case diagram is presented in Figure 1.

Use cases are divided to different sections, grouping logically similar use cases together.

Use case format:

UC0: Use case identifier and title

[POSSIBLY TEMPORARY NOTE HERE FOR PROJECT GROUP; SHALL BE REMOVED FROM FINAL VERSION.]

Scenario A	First scenario for doing the use case.
Scenario B	Second scenario for doing the use case. ...
Precondition	Preconditions for use case.
Postcondition	Postconditions for use case.
Error condition	Error handling, mainly if anything special needs to be done.
Goal-derived	Goal-derived use case(s) in which this use case occurs (if any); see section 6.1.
Requirements	Requirement(s) from which this use case derives; see section 4.

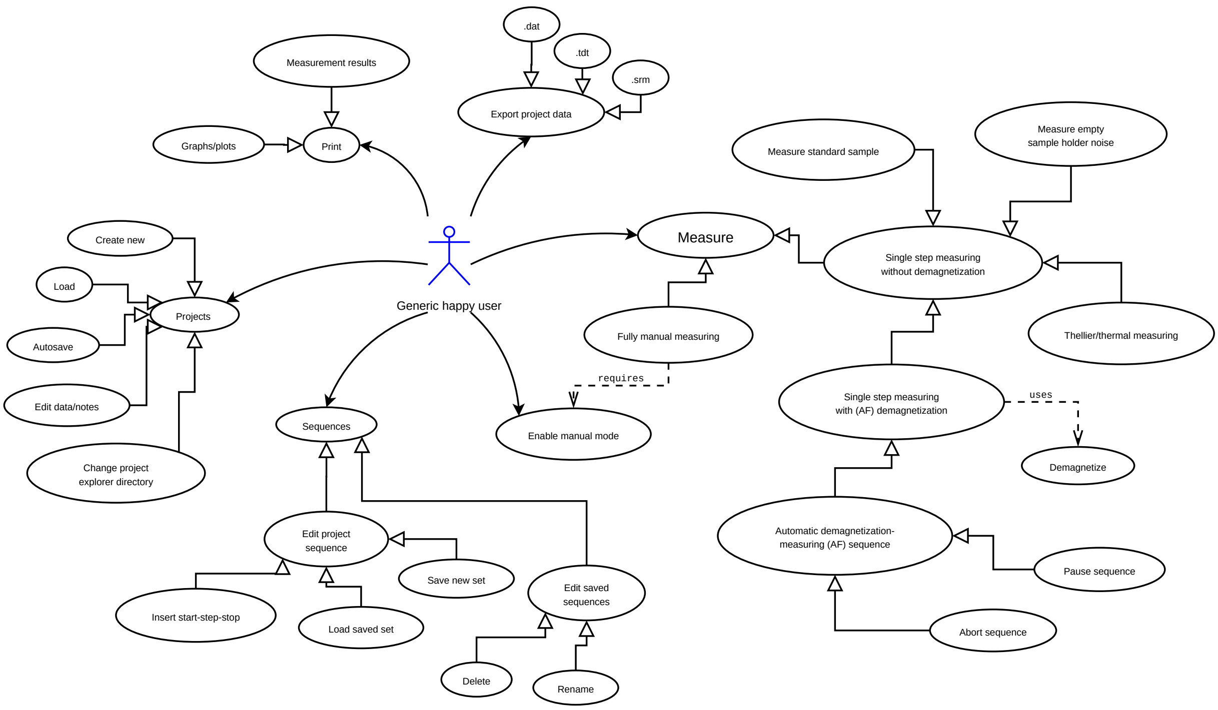

Figure 1: Use case diagram

3.1 Measuring

UC1: Single step measuring without demagnetization

Scenario A	Enter as next AF demagnetization step "0" or empty (default for new projects), meaning no demagnetization, and click "Single step".
Precondition	AF project open, sample in sample holder.
Postcondition	Sample measured, results on screen and appended to current project.
Error condition	The program shall let the user know if something went wrong.
Goal-derived	6.1.3, first few steps.

UC2: Single step measuring with (AF) demagnetization

Scenario A	Enter as next AF demag step anything greater than zero, and click "Single step".
Precondition	AF project open, sample in sample holder.
Postcondition	Sample demagnetized (possibly ruined) and measured, results on screen and appended to current project.
Error condition	The program shall let the user know if something went wrong, and, should the demagnetization field not be coming down, warn user with an alarm sound :)

UC3: Automatic demagnetization-measuring sequence (AF sequence)

Scenario A	Enter the AF sequence (see section 3.5 for ways to enter it) and click "Measure".
Precondition	AF project open, sample in sample holder.
Postcondition	Sample demagnetized according to entered AF sequence (possibly ruined) and measured after each demagnetization, results on screen and appended to current project.
Error condition	The program shall let the user know if something went wrong, and, should the demagnetization field be uncalm, warn user with an alarm sound x)
Goal-derived	6.1.1 whole sequence, 6.1.3 sequence after few single steps at the beginning.

UC4: Pause automatic measuring sequence

Scenario A	While measure sequence is running, click "Pause".
Precondition	Ongoing measuring sequence (UC3).
Postcondition	Measure sequence halts after current step is done, results on screen.
Error condition	Program tells if sequence can't be paused (and something has gone terribly wrong).
Goal-derived	6.1.3, Tomas pauses the sequence as the meteor demagnetizes too rapidly.

UC5: Abort automatic measuring sequence

Scenario A	While measure sequence is running or paused, click "Stop immediately".
Precondition	Ongoing or paused measuring sequence (UC3).
Postcondition	Measure sequence halts immediately [and program enters "fully manual" mode?]
Error condition	Program tells if sequence can't be aborted (and something has gone terribly wrong).

UC6: Thellier measuring

Scenario A	Click "Single step". (Temperature can be entered later, as it won't affect measuring.)
Precondition	TH project open, sample in sample holder.
Postcondition	Sample measured, results on screen and appended to current project.
Error condition	As usual.
Goal-derived	6.1.2, temperatures already entered, one step at 380 °C.

UC7: Thermal measuring

[EXACTLY THE SAME AS THELLIER?]

Scenario A	Click "Single step". (Temperature can be entered later, as it won't affect measuring.)
Precondition	TH project open, sample in sample holder.
Postcondition	Sample measured, results on screen and appended to current project.
Error condition	As usual.

UC8: Measure magnetometer ground noise

[2005-02-23 NOT IN CURRENT UI PROTO, NOR PLANNED FOR IMPLEMENTATION.]

Scenario A	Click "Ground noise" and "Calibrate".
Precondition	No ongoing measurements.
Postcondition	Ground noise measured, results on screen and appended to "Ground noise" project.

UC9: Measure empty sample holder noise

Scenario A	Click "Holder noise" and "Calibrate".
Precondition	No ongoing measurements, empty sample holder.
Postcondition	Holder noise measured, results on screen and appended to "Holder noise" project.
Goal-derived	6.1.1, calibration at the beginning.

UC10: Measure standard sample

Scenario A	Click "Standard sample" and "Calibrate".
Precondition	No ongoing measurements, standard sample in sample holder.
Postcondition	Standard sample measured, results on screen and appended to "Standard sample" project, and, predefined .std file.
Postcondition	Holder noise measured, results on screen and appended to "Holder noise" project.

UC11: Fully manual measuring

[2005-02-25 STILL NOT CLEAR HOW FULLY MANUAL IS SUPPOSED TO WORK; TOMAS WILL BE BACK ON TUESDAY 1.3. AND MIGHT TELL US.]

Scenario A	Click any of the manual control components.
Precondition	Manual mode enabled.
Postcondition	Manual action done, result on screen.

UC12: Enable manual mode

Scenario A	Click "Manual" checkbox above the manual control components.
Precondition	No ongoing measurement.
Postcondition	Manual mode enabled.

3.2 Exporting

UC13: Export project data into .dat file

Scenario A	In project explorer file list, right-click on desired project file, click "Export .dat in current directory".
Scenario B	In project explorer file list, right-click on desired project file, click "Export .dat to disk drive A:".
Scenario C	In project explorer file list, right-click on desired project file, click "Export .dat...", chooce directory and filename to export.
Scenario D	In project explorer file list, select multiple files with shift-click and ctrl-click, and make any of above actions.
Precondition	At least 1 project file in current (selected) directory.
Postcondition	Project data exported to .dat file.
Error condition	Notify if file error occurs (such as no disk in A: drive).

UC14: Export (thellier) project data into .tdt file

Scenario A	In project explorer file list, right-click on desired project file, click "Export .tdt in current directory".
Scenario B	In project explorer file list, right-click on desired project file, click "Export .tdt to disk drive A:".
Scenario C	In project explorer file list, right-click on desired project file, click "Export .tdt...", chooce directory and filename to export.
Scenario D	In project explorer file list, select multiple files with shift-click and ctrl-click, and make any of above actions.
Precondition	At least 1 project file in current (selected) directory.
Postcondition	Project data exported to .tdt file.
Error condition	Notify if file error occurs (such as no disk in A: drive).

UC15: Export single measurement details into .srm file

Scenario A	In measurement result table, right-click on desired measurement line(s), click "Export .srm in current directory".
Scenario B	In measurement result table, right-click on desired measurement line(s), click "Export .srm to disk drive A:".
Scenario C	In measurement result table, right-click on desired measurement line(s), click "Export...", chooce directory and filename to export.
Scenario D	In measurement result table, select multiple lines with shift-click and ctrl-click, and make any of above actions.
Precondition	At least 1 measurement result in current project.
Postcondition	Measurement details exported to .srm file.
Error condition	Notify if file error occurs (such as no disk in A: drive).

3.3 Printing

[2005-02-25 not in current UI prototype, implementation priority low.]

UC16: Print measurement results

Scenario A	Click "Print...", "Measurement results".
Precondition	Open project.
Postcondition	Measurement results printed via [Java] standard printing window.
Error condition	Let know if printing error occurs.

UC17: Print graph sheet (with 7 different graphs; described elsewhere)

Scenario A	Click "Print...", "Grap sheet".
Precondition	Open project.
Postcondition	Measurement results printed via [Java] standard printing window.
Error condition	Let know if printing error occurs.

3.4 Projects

As in project files, which store all measurement results.

UC18: Automatically save all measurement cycles in project file

[PROPABLY MORE OF A REQUIREMENT THAN USE CASE AND SHALL BE REMOVED FROM HERE.]

Scenario A	Make any measurement action (see section 3.1 for those).
Precondition	Project file open.
Postcondition	After measurement (step) is done, new results appended to project file.
Goal-derived	6.1.1, 6.1.2, 6.1.3.

UC19: Create new project

Scenario A	Click the empty line below filenames in project explorer, enter new project name, chooce AF/TH (thellier) project, click "Create new" or press enter.
Precondition	Project explorer in desired directory (UC21).
Postcondition	New project created and selected.
Goal-derived	6.1.1 after calibration, 6.1.3 at the beginning.

UC20: Load project

Scenario A	Click any filename in project explorer.
Precondition	Project explorer in desired directory (UC21).
Postcondition	Existing project loaded and selected.
Goal-derived	6.1.2 all samples already as project files.
Scenario B	Click "Browse..." in project explorer, use standard [Java] file chooser to select project file to be opened.
Precondition	None.
Postcondition	Existing project loaded and selected, project explorer in opened project’s directory.

UC21: Change project explorer directory

Scenario A	Click into current directory textbox in project explorer, write directory to change to, press enter.
Scenario B	Click into current directory textbox in project explorer, start writing directory; when autocomplete-results appear below, use up/down+enter or mouseclick to select the directory.
Scenario C	Click the down-arrow on the right side of current directory textbox in project explorer, choose desired directory from appearing directory history.
Scenario D	Click "Browse..." in project explorer, use standard [Java] file chooser to select directory to change to.
Precondition	Project explorer in desired directory.
Postcondition	New project created and selected.
Goal-derived	6.1.1, 6.1.2, 6.1.3.

UC22: Insert/edit project data

Scenario A	Click any of the project data checkboxes, or textboxes and enter value.
Precondition	Open project (usually just created).
Postcondition	Project data changed, saved automatically to project file.
Goal-derived	6.1.1 after creating new project.

3.5 Sequences

As in automatic demagnetization-measuring sequences (AF sequences), or, thellier temperature sequences.

UC23: Insert sequence with start-step-stop values

Scenario A	Click "Start" textbox, insert start value, click "Step" textbox, insert step value, click "Stop" checkbox, insert stop value, click "Add sequence" or press enter.
Scenario B	Click "Start" textbox, insert start value, press tab, insert step value, press tab, insert stop value, press enter or click "Add sequence".
Precondition	An open project.
Postcondition	Sequence from *start* to *stop*, increasing by *step* for every step, appended to measurement result table, and selected.
Goal-derived	6.1.3 after creating new project.

UC24: Load sequence

Scenario A	Click down-arrow in right side of "Load set" combo box; from appearing list, click sequence to load.
Precondition	An open project, at least 1 saved sequence.
Postcondition	Selected sequence appended and selected to measurement result table.
Goal-derived	6.1.1, after typing project data, load "Basalt" set; 6.1.2 set already loaded for each project file (when created).

UC25: Edit sequence on-the-fly

Scenario A	Click any unmeasured "Tesla" or "Temp" column in measurement result table, enter new value.
Scenario B	Click any unmeasured line in measurement result table, press del to delete that line.
Scenario C	Right-click any unmeasured line in measurement result table, choose "Delete" to delete that line.
Scenario D	Click-drag any unmeasured line in measurement result table to new position within unmeasured lines.
Scenario E	Select multiple unmeasured lines in measurement result table with shift-clicks and ctrl-clicks, make any of above actions (except enter new value).
Precondition	Unmeasured lines in current project's measurement result table (note that measuring sequence can be going on).
Postcondition	Editing committed.
Error condition	Editing or screwing up measured lines won't be allowed.
Goal-derived	6.1.3, when sequence demagnetizes too rapidly, Tomas deletes unmeasured lines.

UC26: Save sequence

Scenario A	Click "Load set" combo box, enter new sequence name, click "Save set". [2005-02-25 not in current (almost final) UI prototype!]
Scenario B	Right-click any line in measurement result table, choose "Save full sequence...", enter name, press enter.
Scenario C	Select any lines in measurement result table with shift-clicks and ctrl-clicks, right-click on selected lines, choose "Save selected sequence...", enter name, press enter.
Precondition	At least 1 line in measurement results table (although you propably don't want to save a sequence with only one step x).
Postcondition	New sequence set saved to predefined sequence file and available from "Load set" combo box.
Error condition	Ask whether to overwrite, if sequence with the same name already exists (allow to enter new name if choose not to overwrite).

UC27: Edit stored sequence

Scenario A	UC24 "Load sequence" -> UC25 "Edit sequence" -> UC26 "Save sequence" with same name as the loaded sequence.
Scenario B	Click menu "Options"->"Sequences...", edit any sequence in appearing window... [2005-02-25 not in UI prototype, must be improvised :)]
Precondition	At least 1 saved sequence.
Postcondition	Changes saved to predefined sequence file.

UC28: Delete stored sequence

Scenario A	Click down-arrow in right side of "Load set" combo box; from appearing list, right-click sequence to delete, choose "Delete".
Scenario B	Click menu "Options"->"Sequences...", delete sequence in appearing window... [2005-02-25 not in UI prototype, must be improvised :)]
Precondition	At least 1 saved sequence.
Postcondition	Sequence deleted, changes saved to predefined sequence file.

UC29: Rename stored sequence

Scenario A	Click down-arrow in right side of "Load set" combo box; from appearing list, right-click sequence to rename, choose "Rename...", enter new name, press enter.
Scenario B	Click menu "Options"->"Sequences...", rename any sequence in appearing window... [2005-02-25 not in UI prototype, must be improvised :)]
Precondition	At least 1 saved sequence.
Postcondition	Sequence renamed, changes saved to predefined sequence file.

4 User requirements

Goals of the software set by client..

4.1 Functional requirements

4.1.1 Magnetometer

Identifier: R1
Name: SQUID control and usage
Description: Able to control Squid-magnetometer and make measurements with it.
Priority: 1

Identifier: R2
Name: Manual control
Description: Able to operate the magnetometer manually. (TODO: what this means?)

Priority: 2

Identifier: R3
Name: Calibration reminder
Description: Magnetometer must be calibrated every 24 hours. The program will remind the user when it would be time to do calibration.
Priority: 2

4.1.2 Project files

Identifier: R4
Name: Create project
Description: Create a new project file which will include the measurement sequence, measurement results and information about the sample. The file format will be custom made for this program.
Priority: 1

Identifier: R5
Name: Autosave project
Description: Program will save measurement data and project information after every measurement step and modification.
Priority: 1

Identifier: R6
Name: Load project
Description: Saved projects can be loaded into program.
Priority: 1

Identifier: R7
Name: Edit project
Description: Ability edit project information and measurement data afterwards. (TODO: which data fields?)
Priority: 2

Identifier: R8
Name: Recalculate derived measurement data
Description: When the measurement data or the mass/volume of the specimen is changed, recalculate all numbers that have been derived from the modified data.
Priority: 2

Identifier: R9

Name: Append project
Description: New measurements can be added to existing projects.
Priority: 2

Identifier: R10
Name: Export to other file formats
Description: Measurements can be saved in .dat, .tdt and .srm files.
Priority: 1

4.1.3 Measurements

Identifier: R11
Name: Create a measurement sequence
Description: Able to create AF and Thellier measurement sequences with several different sized steps.
Priority: 2

Identifier: R12
Name: Automatic sequence
Description: Do all the measurement steps automatically.
Priority: 2

Identifier: R13
Name: Single step sequence
Description: Do only one measurement step at a time.
Priority: 2

Identifier: R14
Name: Pause automatic sequence
Description: Able to stop the measurement sequence after current step. After being stopped, the user can make modifications to the sequence and continue from where the sequence was left.
Priority: 2

Identifier: R15
Name: Panic abort
Description: Able to stop any measurement immediately. The demagnetizer will be turned off and the sample holder will stop where it is. The current measurement data will be discarded.
Priority: 1

Identifier: R16
Name: Numeric presentation of measurements
Description: Program shows measurement data in numbers. Measurement data will be displayed using scientic notation (1.23e4).
Priority: 1

Identifier: R17
Name: Graphic presentation of measurements
Description: Program draws graphs from measurement data. Available graph types are: Stereo Plot, Intensity, Zijderveld, Difference Vectors, Susteptibility, O63, Great Circles. (TODO: what are the definitions of these graphs? priorities per graph?)
Priority: 4

4.1.4 Others

Identifier: R18
Name: Hotkeys
Description: The program has hotkeys for the most often used operations.
Priority: 3

Identifier: R19
Name: Change hotkeys
Description: Possibility to change hotkey assignments.
Priority: 4

Identifier: RXX
Name:
Description:
Priority:

4.2 Quality requirements

Identifier: QR1
Name: Accuracy of measurements
Description: The calculations that the program makes from measurements, must be done right.
Priority: 1

Identifier: QR2
Name: Ease of use

Description: Program should be easy to use for first time users after it has been explained to them how to take measurements with the magnetometer.
Priority: 1

Identifier: QR3
Name: User guide
Description: Program should have good help pages.
Priority: 3

Identifier: QRXX
Name:
Description:
Priority:

4.3 Environment

The computer that will run this program will be equal or better to 1GHz CPU, 256MB RAM, 1280x1024 resolution. The program must run under Windows XP. The hardware and communication with it is described in the section "External interfaces".

4.4 Maintainability

It must be possible to continue the development of the program. The documentation must be complete so that other teams can quickly continue the development even if they have not studied user interface designing. It must be possible to add new graph types to the program and export the measurement data to other programs.

4.5 Restrictions

Program will be used in normal PC which is connected to magnetometer. Taking into account rapid phase of computer evolution it is possible that computer in which program is used can change, accordingly the program should be able to be installed by outsiders. We need not prepare to changing of magnetometer, as new magnenetometer will probably have its own program.

5 System requirements / Functions

Specific explanation of the functions to be implemented

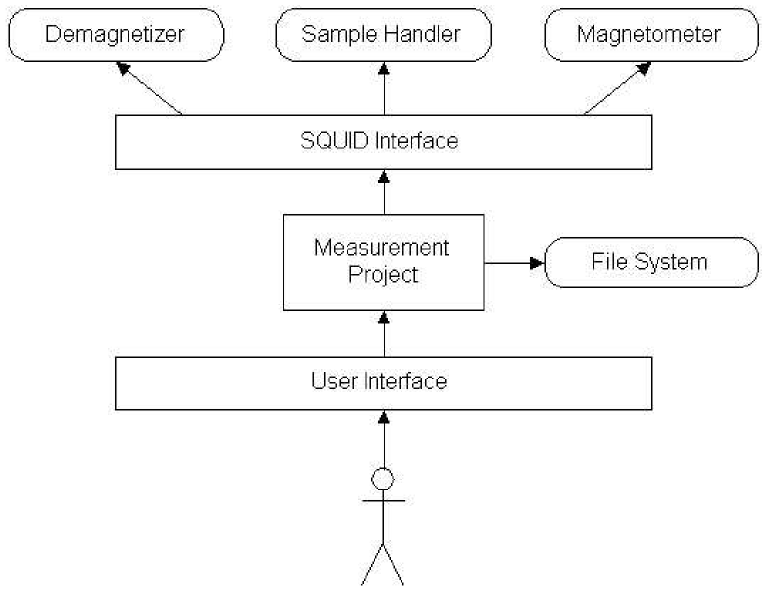

Figure 2: Architecture overview

5.1 System restrictions

6 User interface

Overview of UI described here..

6.1 Goal-derived use cases

6.1.1 Goal-derived use case 1: Erkki...

6.1.2 Goal-derived use case 2: Fabio...

6.1.3 Goal-derived use case 3: Tomas...

7 Architecture overview

The program architecture can be divided into three parts: SQUID interface, measurement project and user interface. The program will also communicate with the SQUID hardware and local file system. A graphical representation of the architecture can be seen in Figure 2.

SQUID interface is responsible for controlling the hardware in an orderly manner. It will provide high- and low-level controls for using the hardware. Communication with the hardware is done via COM ports. The SQUID interface will hide the protocol-level commands from the programmer and prevent illegal use of the hardware.

Measurement project is responsible for managing the project information, measurement sequence and measurement data. It will recieve commands from the user interface and notify the UI when the state of the project changes. It will send commands to the SQUID interface, recieve measurement data and save it. When the internal data of the project changes, the copy on the local file system will be automatically synchronized after a short delay (1 second or less).

User interface is responsible for communicating with the user of the program. It will update itself whenever the state of the measurement project is changed. It will send commands from the user to the program.

8 External interfaces

Interfaces to existing software and hardware are described here.

8.1 Existing program

The existing software for using the SQUID is "2G Enterprises Data Acquisition". We have access to the source code for version 2.99.3 of the program. From the old source code we will reuse basically only the SerialIO component. We will build an interface for communicating with the SQUID hardware by using Java and JNI (Java Native Interface).

8.2 Hardware control protocols

The SQUID consists of three independent units:

- Automated Sample Handler System (MODEL 2G800)
- Automatic Sample Degaussing System (MODEL 2G600)
- Superconducting Rock Magnetometer (MODEL 755R or 760R)

Automated Sample Handler System controls the movement and rotation of the sample holder. Its protocol is described in Appendix 1.

Automatic Sample Degaussing System controls the demagnetizer. Its protocol is described in Appendix 2.

Superconducting Rock Magnetometer reads the measurements from the magnetometer. Its protocol is described in Appendix 3.

9 Validation

Description of how to validate the set requirements.

Appendix 1. Automated Sample Handler System Protocol

Korvaa tämä sivu tiedostolla "Automated Sample Handler System - Protocol.pdf"

Appendix 2. Automatic Sample Degaussing System Protocol

Korvaa tämä sivu tiedostolla "Automatic Sample Degaussing System - Protocol.pdf"

Appendix 3. Superconducting Rock Magnetometer Protocol

Korvaa tämä sivu tiedostolla "Superconducting Rock Magnetometer - Protocol.pdf"