Software Requirements Specifications

Version 1.0

SOURCERER
(A Sourcemeter Magician)

Client:-
Dr. Somaditya Sen

Developed by :-
1.Amey Kiran patel (160001003)
2.Akash Ahirwar(1600010053)
3.Chinmay Anand(160001018)
4.Sandarbh Sahu (160001053)
5.Mohit Malviya (1600001035)

Prepared for CSE 258 Software Engineering-Spring 2018

Revision History:-

Date	Description	Comments
26/01/2018	Version 1	Completed the SRS as per the client's requirements as of 25/01/2018.

Document Approval:-

Signature	Title	Date

Table of Contents

1. Introduction

This SRS (Software Requirements Specification) provides a complete idea on how to design and implement the software as well as a basic idea of its usage and benefits. It provides a complete idea of the classes of users who will be using this software and how they will be benefited by it.

1.1 Purpose

The SRS (Software Requirements Specification) gives a comprehensive idea on the design, implementation and usage of the software. This SRS document provides a detailed overview of our software product, its parameters and goals.This document describes the project's target audience and its user interface, hardware and software requirements.
Any software developer can design and implement the software using this document. Also the regular users can get a basic understanding of its usage.

1.2 Document Conventions

Terms	Definitions
Sourcemeter	An electronic device with sensitive measurement and precise sourcing capabilities.
SRS	Software Requirements Specification
SCPI	Standard Commands for Programmable Instruments
GUI	Graphical User Interface
DUT	Device Under Test

Keithley Instruments	It is a measurement and instrument company that developed, manufactured, marketed, and sold data acquisition products, as well as complete system solutions for high-volume production and assembly testing.
NI	National Instruments,producer of automated test equipment and virtual instrumentation software.
LabView	Laboratory Virtual Instrument Engineering Workbench, is a programming environment in which you create programs using a **graphical** notation.
PID Controller	Proportional–Integral–Derivative Controller
IDE	Integrated Development Environment
OS	Operating System

1.3 Intended audience and reading suggestions

This SRS is developed with a major aim of guiding the creation of a software that shall assist Dr. Somaditya Sen in his ongoing project to study the electrical and thermal properties of a material using Keithley Sourcemeter, however it can be used by all Keithley Sourcemeter users to perform some basic measurements
like measuring current-voltage-time variables and plotting their graphs accordingly.

The software developers with a desire to understand the design and implementation of our project can refer to further sections of the SRS for better understanding of how the software is supposed to be created and the vision behind its creation.

1.4 Software Scope

The purpose of the software ('Sourcerer') is to connect the Keithley Sourcemeter with a pc and perform some basic measurement experiments using a simple GUI (without having to touch the device manually).

Particularly, user can supply a constant voltage (whose magnitude can be chosen by the user) to a sample and record the current varying with time.

Also, user can change light intensity (using ON/OFF state of a light source) falling on the sample with desired time period. Then he can measure current and resistance with time which will help to analyse the behaviour of sample with changing light conditions when constant voltage is applied across it .

User will not only obtain readings but will also be able to plot graph of current vs time , voltage vs time etc. for a better understanding. They can record all their observations for future reference (they will not have to perform the same experiment all over again).

1.5 References

- pyVISA Docs - https://media.readthedocs.org/pdf/pyvisa/latest/pyvisa.pdf

- Keithley Sourcemeter 2400 Series Manual - http://docs-europe.electrocomponents.com/webdocs/10eb/0900766b810eb7d6.pdf

- PID Controller Videos -
  - https://www.youtube.com/watch?v=UR0hOmjaHp0
  - https://www.youtube.com/watch?v=0vqWyramGy8

- Thermocouple Docs - https://www.thermocoupleinfo.com

- Cryostat Docs - http://abbess.com/cryo/cryostat-cold-fingers-and-sample-holders/

- PDF subsuming commonly used SCPI commands- http://literature.cdn.keysight.com/litweb/pdf/E4400-90506.pdf

2.General Description

2.1 Product Perspective

The programming of a measuring device can be vexing at times, notably when you are a novice. There does exist powerful software namely LabView which does ease the transition, however the software can prove to be surprisingly bothersome with the computer requiring numerous drivers and subsisting software. Not everyone is able to obtain the license to run the LabView. The powerful software requires a good processor and consumes lot of disk memory, hence draining a lot of resources too.

Therefore, the client wishes the developers to create a powerful,low-cost and easy to use software to serve his purposes.Using simple yet powerful means, the client wishes to link his computer to the measuring device (Keithley Source-meter Model 2401) and instruct it to behave accordingly.

2.2 Product Functions

The following are the aims the projects serves to achieve in order to meet the client's requirements :-

1)

- ❏ To automate a Keithley Source-meter Model 2401 and hence to enable its users to command the device to provide a constant voltage for a required time period without the need of manually setting up the device to do so.
- ❏ To record the voltage-current-time variables from the Keithley source-meter and plot it on a graph accordingly.

2)

- ❏ To allow users to pulse a stimuli source creating ON/OFF states for a desired time period in order to observe sensitivity to the particular stimuli from changes in resistivity of samples during the ON/OFF states.
- ❏ To record voltage-current-time variables from the Keithley output on two/four probe setups already available to users and from it calculate sensitivity of samples and provide as output.

3)

- ❏ To design a complex heating system to control the temperature of samples at desired temperature and to be able to regulate the temperature using the regulation of current of the heater.
- ❏ To create a working physical device based on design proposed above.

2.3 User Classes and Characteristics

The software is open sourced and coupled with the unavailability of any manual on the internet that guides a novice programmer on how a source-meter (For example, Keithley 2401) can be programmed as per the developer's needs means the software being created is bound to captivate a large section of the community wishing to achieve the same.

The classes expected to be lured into using the software can be classified as:

1. **Professors**

   The software created helps in studying and analysing the electrical and thermal conducting properties of the DUT (Device Under Test).

   The interface created provides a user-friendly GUI which would not require them to know about the underlying lines of codes required to create the software.

2. **Developers**

   The software with its underlying lines of codes provides the developers with a protocol to follow to quickly understand how any kind of measurement device can be controlled independently of the interface (For example, GPIB, Ethernet, Wi-Fi, RS232, USB).

3. **Engineers**

   The electrical engineers, at the end of the day, have a software that allows them to automate the trivial tasks (For example, switching on and off of the device, applying constant voltage, providing impulses in regular or intermittent intervals) and obtain results of experiments that can be safely stored in a computer, reducing human labour.

   The mechanical engineers have a cryostat which permits them to measure resistivity of substances at low temperatures (say, -173^{o} C). Having used PID (proportional integral derivative), we enable them to maintain such constant low temperature for prolonged durations. This otherwise is very taxing and difficult to achieve manually.

2.4 General Constraints

- The software is expected to perform real time measurements of voltage,current and resistance so it is to be developed keeping that in mind.
- The 2400 SourceMeter Series includes instruments with voltage source and measure capabilities ranging from 200V to 1100V. The minimum voltage source step size is 5uV and the maximum measurement resolution is 1uV so the precision of measurement would not be more than 5uV.
- The SourceMeter series products have maximum current ranges of 1A, 3A, 5A and 10A (pulse only). The base Model 2400 has a minimum current source step size of 50pA and a maximum measurement resolution of 10pA.
- The client needs a graphical based interface for easy and convenient operation of the device while carrying out sensitive and concentration demanding experiments.

2.5 Assumptions and Dependencies

To perform the first part of the project, it is assumed that the drivers that will be mentioned later on have been installed. The user is expected to have basic knowledge of handling the measuring devices.

To carry out the second part of the project, we need a function generator to provide pulse stimuli signals creating ON/OFF states for the required time period, or we can program an **Arduino** to send stimuli for signals which is a more intriguing way of doing the same.

To perform the third part of the project, the user is expected to have a cryostat (which we have constructed in our workshop), liquid nitrogen in a dewar flask, a heater (which can be controlled by changing current supply).

3. External Interface Requirements

3.1 User Interfaces

- ➢ The interactions of the electronic device is to be carried out through graphical icons and visual indicators, instead of a text-based user interface, typed command labels or text navigation.
- ➢ Thereby, a simple GUI (Graphical User Interface) is to be created which enables the user of the “ Sourcerer ” to access every feature provided to them by the software with just a few clicks and text entries .
- ➢ User will provided with 2 options. The first option would allow controlling of the sourcemeter and the second would allow that of the arduino .

3.2 Hardware Interfaces

- ➢ The KUSB-488B from Keithley is an IEEE-488 USB to GPIB interface converter. The USB-to-GPIB interface turns a computer with a USB port into a fully functional GPIB controller.

The Model KUSB-488B USB to GPIB converter has the following advanced features:

- ❏ IEEE-488.1 and IEEE-488.2 standards compatible.
- ❏ Plug and play interface (USB 2.0 compatible).
- ❏ Built-in 32KB first-in first-out buffer for read/write operations.
- ❏ No external power required.
- ❏ Device drivers for Microsoft Windows operating systems which can be downloaded from https://www.tek.com/accessory/kusb-488b-software.
- ❏ Command-compatible driver APIs (application program interfaces) for use with Keithley Instruments, National Instruments(NI) and VISA libraries.

- ➢ The Keithley Source meter uses two remote interface ports (IEEE-488/GPIB and RS-232C).

- ➢ The thermocouple senses the temperature of the sample and supplies it as a voltage signal to the PID controller.

3.3 Software Interfaces

1. VISA is a standard for configuring, programming, and troubleshooting instrumentation systems comprising GPIB, VXI, PXI, Serial, Ethernet, and/or USB interfaces. The NI-VISA can be downloaded from http://www.ni.com/download/ni-visa-17.5/7220/en/

2. The VISA specification has explicit bindings to Visual Basic, C, and G (LabVIEW's graphical language). However, you can use VISA with any language capable of calling functions in a shared library (.dll, .so, .dylib). PyVISA is Python wrapper for such shared library.

3. PyVISA is a Python frontend for the VISA library that enables controlling all kinds of measurement equipment through GPIB, RS232, USB and Ethernet among others interfaces.

4. The SCPI (Standard Commands for Programmable Instruments) defines a standard for syntax and commands to use in controlling programmable test and measurement devices.

5. The physical hardware communications link is not defined by SCPI. While it was originally created for the IEEE-488.1 (GPIB) bus, SCPI can also be used with RS-232, RS-422, Ethernet, USB, VXIbus, HiSLIP, etc.

6. We can send SCPI commands to Keithley source meter with the help of PyVisa.

3.4 Communication Interfaces

- Computer communicate with Keithley source meter through KUSB-488B USB to GPIB converter device.

4. System Features

4.1 Automation of the Source-meter to provide constant voltage

4.1.1 Description and priority

As the title suggests, the software will automate the source-meter performing tasks without manual labour. It will enable application of constant voltage (as per the user's requirements) and study the properties of substance and declare if the substance is ohmic or non-ohmic in nature. We can deduce if the substance is an insulator or conductor too and preserve this data in a database which can further be used elsewhere.

This feature is of a high priority as the current vs time graph of the sample plays a crucial role in determining its properties.

4.1.2 Stimulus/Response Sequences

➢ The user will first switch ON the device by clicking a button present in the GUI presented to the client. This will be confirmed by the activation of the device indicated by a blue light in case of the Keithley source-meter.

➢ The user should then provided with a menu that allows him to choose what he desires to perform. For this project, we shall choose to apply constant voltage to the DUT.

➢ After choosing the aforementioned option, the user will be asked to enter the time duration for which the voltage is to be applied.

- ❏ Scenario 1: If the user enters a value which is reasonable, it will proceed ahead asking the user for further information.
- ❏ Scenario 2: If the user enters a value which is unreasonable (say 2 years), it will issue a warning asking the user to re-enter an appropriate value to carry out the execution.

- ➢ After putting down the time duration of the event, the user will be asked to enter the value of the voltage to be applied across the DUT.
  - ❏ Scenario 1: If the user enters a value which is safe to use, the software will execute the request and obtain the values in a database will be saved in the computer .This data is used to provide the user with a graph which provides further details on the properties of the DUT.
    Based on the properties of the curve obtained on the graph, we assert a few properties of the substance.

  - ❏ Scenario 2: If the user enters a value which is deemed to be unsafe, the user will be alerted and the execution will be halted with the user being taken back to initial menu. This is a precautionary measure as high voltages could result in non-recoverable hardware failures.

- ➢ The user is provided with an option to save the data obtained which when used, asks the user, “Do you wish to save the experimental values obtained?”.
  - ❏ Scenario 1: If the user opts to save the experimental results, the values are stored in a database on the computer.
  - ❏ Scenario 2: If the user chooses “NO”, he is asked once again for confirmation the database file is deleted from the computer.

- ➢ Finally, the user is then returned to the main screen to again begin with.

4.2 Testing a Light-sensitive device by programming an arduino

4.2.1 Description and Priority

We are supposed to send pulses of light to the given sample which, being photosensitive, shows changes in resistivity which we are supposed to measure using the Keithley source meter again or we can also measure this change using a four probe apparatus or even a two probe apparatus. We power our Arduino device by connecting it to our laptop or computer or main system using Arduino connector and command it to send pulses due to which we continuously switch on and off the LED connected to it, thus generating pulses of light which would alter the resistivity of the sample and hence we would be measuring it using the Keithley Source meter again. We can control the functioning of Arduino using either Arduino IDE software or the Arduino library of Python.

4.2.2 Stimuli/Response Sequence

- The user first activates the arduino by clicking a button.
- The user is asked about the intensity of light he wishes to obtain from the light emitting device. Based on the user's chosen option, the arduino instructs the LED to produce the type of intensity to be produced.
- The user then sets the voltage required off the source-meter by following the same process as the guided by the first system feature.

4.3 Enabling Calculation Of Resistivity at Very Low Temperature

4.3.1 Description and Priority

Our setup includes a heater placed in a cryostat, put in contact with the sample. The heater is a resistive one. So when we pass current through it, it generates heat. In order to compensate for the magnetic field produced due to current flow in the coil wound over the heater, we roll another coil over it but this time making sure that this winding compensates for the field produced due to the former one. We have a thermocouple that senses the temperature of this sample now and supplies it as a voltage signal to the PID controller with the help of which we maintain its temperature at a constant value thus enabling the user to take readings and perform experiment on the sample even in extreme temperature conditions, as in 77K of the Nitrogen dewar in this case. The thermocouple provides interface for the conversion of temperature to voltage and the PID controller further enhances this interface by manipulating the current flow and thus enabling constant temperature conditions for the sample.

The device constructed consists of its own PID circuit and sensors to record the temperature changes and act accordingly.

4.3.2 Stimulus/Response Sequences

- The user will be asked about the temperature at which the resistivity is to be calculated.
    - Scenario 1: If the user requests to carry out measurements at temperatures which the device fails to measure at, it issues a warning and fails to execute.
    - Scenario 2: In case the value entered is appropriate, the experiment is carried out and results are displayed.

5. Non-Functional Requirements

5.1 Performance

The software is created mainly using python3 and makes use of its various powerful in

built modules like pyvisa, matplotlib, Arduino, tkinter, csv, sqlite3, etc. These modules are well-optimized and provides for quick and accurate executions.

We provide for a user-friendly GUI which does not hinder the performance of the software, hence ensuring high performance.

5.2 Availability

The software being created is readily accessible to all its potential seekers as it will be published on the internet. However, it requires the installation of certain drivers and python modules to be functional. These requirements have already been written about in Section 3 and should be revisited for further clarity.

5.3 Security

The data obtained from the results of the experiments is safely stored in the user's computer. As long as the data on the user's computer is secure, the software remains uncompromised.

5.4 Safety

The Keithley source-meter in use for the project in itself is a very safe device. However, it is not recommended to apply high voltages across the device under test as it might lead to unavoidable loss in data and failure of system hardware.

It is advisable to use DC current as Arduino works only at DC (Direct current). A voltage greater than 12 V can damage the device itself.

5.5 Maintainability

The software is created by collaborating using a free and open source distributed version control system, Git, and hence contributions of various other programmers and thereby remains updated.

Competent documentation should assist a new programmer to understand the functionality added by each code and the goals we aspire to achieve.

5.6 Portability

"Sourcerer" runs on Python3's IDE. As it has an IDE of its own,it is super portable and can be run on any platform or any OS.

5.7 Logical Database Requirements

We shall use databases to store the results of experiments on our computers so that it can be easily altered, managed and retrieved. It should incorporate a simple table consisting of attributes like resistance, current and voltage corresponding to a particular time interval after the start of the experiment.