VISA Communication Tool Requirements Documents

Ian Absher[1], Seth Ward[1], Chenliang Wang[2], Lucien Armand Tamno[2] and Wenbo Hou[2]

[1]Tektronix, Inc.
[2]Oregon State University

November 2016

Abstract

This document aims to provide detailed information about the project, VISA Communication Tool. It will provide project overview and interface designs. Specific requirements and design constrains for the develop team also exist in this paper. At the end of this document, the develop team will specify each team member's work and the Gantt Chart to show the develop pipeline.

Contents

1 Introduction

This section briefly describes the content of this SRS document. The purpose of this document and necessary abbreviations/definitions are also in this section.

1.1 Purpose

This document aims to provide a detailed description of the requirement for the "VISA Communication Tool" (VCT) software. It will illustrate the objectives and development of the software,including the overview of the software, the design constraints, various interfaces and the baseline for any future interactions.

1.2 Scope

VCT is a Python-based VISA communication tool that manages test & measurement equipments. Users use this tool to deal with large amount data and locate a device in the network. Additionally, the software will support users with syntax reminders that can complete incomplete commands automatically and highlight incorrect command syntax.

1.3 Definitions, acronyms, and abbreviations

Table 1: Definition

Terms	Definition
API	a set of subroutine definitions, protocols, and tools for building software and applications
GPIB	General Purpose Interface Bus [1]
I/O	Input/Output—The communication between an information processing system
IP address	Numerical labels of devices in a computer network that uses the Internet Protocol for communication
ISO9421	A multi-part standard for Standardization covering ergonomics of human-computer interaction. [2]
OSI	Open systems interconnection model
TCP/IP	The Transmission Control Protocol (TCP) and the Internet Protocol (IP), most common networking protocols
Tektronix	Test & Measurement Company
Users	Engineers who use Tektronix product
VISA	Virtual Instrument Software Architecture
visa32.dll	VISA API library designed for Windows DLL [3]
VCT	Name of the software
VXI	VME eXtensions for Instrumentation. Open standard platform for automated test based upon VMEbus [4]

1.4 Reference

[1] National Instruments Corporation, "GPIB Instrument Control Tutorial," in National Instruments, 2015. [Online]. Available: http://www.ni.com/tutorial/2761/en/. Accessed: Nov. 03, 2016.

[2] International Organization for Standardization, "ISO 9241-210:2010," in ISO, 2015. [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075. Accessed: Nov. 03, 2016.

[3] J. Smith, "visa32.dll File information & Error Howto Fix Guide," in RegistryNuke, 2011. [Online]. Available: http://registrynuke.com/dll-info/visa32-dll-errors-fix. Accessed: Nov. 1, 2016.

[4] I. Poole, "VXI Technology Tutorial," in radio-electronics.com. [Online]. Available: http://www.radio-electronics.com/info/t_and_m/vxi/vxi.php. Accessed: Nov. 1, 2016.

1.5   Overview

There are three more chapters remaining in this document. The second part of this document intend to fully describe the software including the design principles, design constraints as well as the target users.while, the third chapter elaborates on specific requirements of the client in both the interface part and the functionality part. lastly,we will end up the document by providing details about the time line and milestones to be reached.

2 Overview description

This section will give an overview of the whole software. The interaction between each part and the functionality of the tool will be mentioned in this part. A summary of software functionalities will be shown in this section. The design assumptions and constraints will be provided at the end of the chapter.

2.1 Product Perspective

This software allows users to manage test & measurement instruments with commands. Users can input instruments to acquire and operate data from different machine. To make it convenient, the software will put syntax support to users.

Basically, this software implements the VISA Communication protocol to interact with measurement instruments. Instruments get orders from VCT, and then send back data to the software. All interactions happens in TCP/IP. VCT will provide a temporary storage for each process to store the unprocessed binary data. So, users can interpret the data to debug. At the end, the software has a built-in command library that is used for the command reminder.

2.2 Product functions

VCT is a control software for test & measurement instruments with both programmatic and graphic interface. Users can use commands, built-in buttons, even keyboard shortcuts to send instructions. Instead of traditional plain text format, VCT will display collected data based on users' choices. For example, VCT can show continuous temperature change of a rod in 3D coordinates. VCT also has a subwindow to show available devices in the network, so that users can easily find device in need. Furthermore, it allows users to locate target device through IP address.

VCT can save users' working time by providing syntax reminders. When users are typing, the software will query commands based on inputting letters. Then, users will get a list of recommended command syntax as a reference. VCT automatically runs syntax check right after users type in a complete command. Incorrect commands will be highlighted.

Users are able to access to the bottom layer of VCT to check whether the collected data is reasonable or not. Error information will be recorded by VCT.

2.3 Constrains

One constraint in this project is the software's compatibility. Since Tektronix has different types of product, the design team may not be able to make VCT works well with all product. Compatibility issues also occurs when VCT works in different operating systems. The develop team need to take care of memory usages for each process to prevent data loss or memory leak. The last constraint is update issue. Both VISA communication protocol and Tektronix product keep changing. If VCT does not follow it, it may lead to inefficiency or even garbage data collection.

2.4 User characteristics

The design team considers that only two kind of users exists for this product. One is engineers who are first time touching test & measurement instruments. They are not familiar with VISA communication protocol, and make mistakes easily. Another type is experienced engineers who know well about VISA. This group of users try to explore more about the data. For example, they want to verify the correctness of collected data or optimize performances of measurement devices.

2.5   Assumption and dependencies

The design team assumes that all Tektronix product works well. The working computer should have enough memory for temporary memory allocation. Since VCT is VISA based, we assume that all APIs (Application programming interface) are also VISA based. The last assumption is that VCT users can understand English.

3 Specific requirements

This section contains all of the requirements of the software. It includes details about each of requirements.

3.1 User interface

The user interface of this software is user friendly and accessible. We provide a tool bar that contains basic operations. Users can click them to find files, check connections, and etc. The main body of the user interface is the programmatic windows where users can type commands in. We also provide click buttons for users to change input mode or check history.
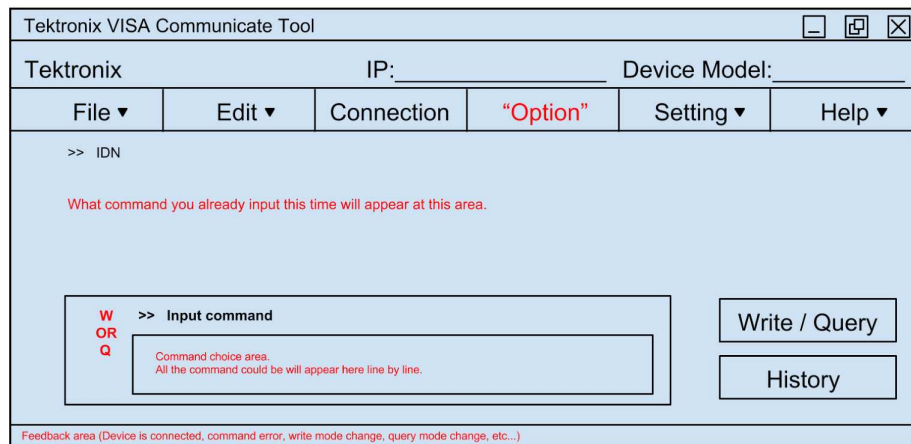

Figure 1: General UI design

As our client mentioned, we will create keyboard shortcuts for major functionalities. For example, users can press Ctrl and S to save collected data. Further more, we will keep improving our UI design as the project goes.

3.2 Hardware interface

Since our goal is to design the management software, no formal hardware interface is needed. However, we will generate a subwindow in the user interface to show status of available measurement instruments.For example, offline devices have a red spot besides, and on-line deices have a green spot besides. Besides that, we will provide general information about each device including, the port number, IP address and etc.

3.3 Communication interface

To communicate with measurement devices, our product implements VISA communication protocol over T&M-specific I/O interfaces such as GPIB and VXI. We will take advantages of visa32.dll that contains VISA APIs in C language to build the communication interface through TCP/IP. We will not process data in this layer, so that users can interrupt it to debug.

3.4 Functional requirements

3.4.1 Programmatic interface

Users can input commands or use keyboard shortcuts to send instructions to devices. Besides inputting commands, users can switch to query mode where they can query commands with different key words.

3.4.2 Command auto-completion

VCT will provide a list of queried commands based on letters user input. As a result, users can manually or use a keyboard shortcut to complete the command with right syntax.
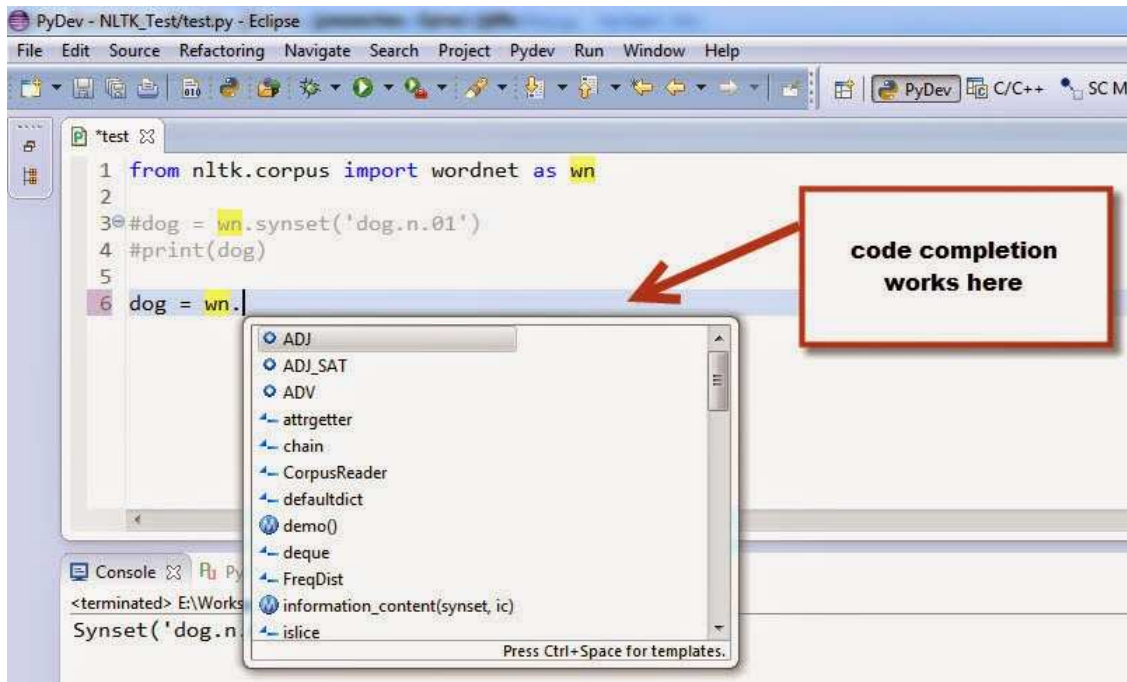

Figure 2: command auto-completion example

3.4.3 Syntax check

When the user type a command with incorrect syntax, the software will underline this command and provide a suggestion to fix it.

3.4.4 Track device

VCT allows users to locate a device that is available in the local network. Users can do that by finding the device in the available device list or enter IP address with finding key words.
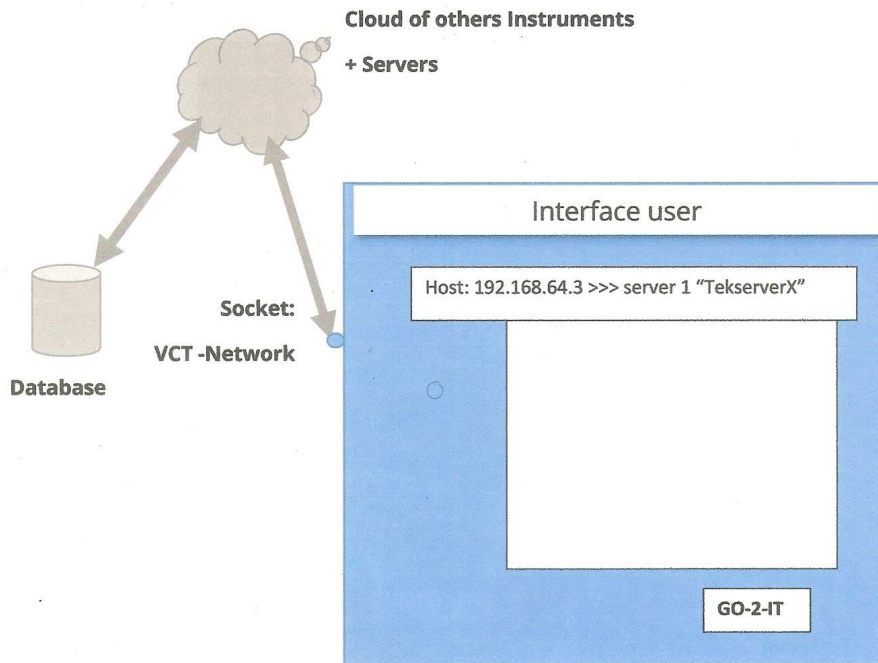
Figure 3: locate device trough IP address

3.5   Software system attribute

3.5.1   Reliability

From the viewpoint of a user and the client, the VCT software has to strictly provide the accurate list of commands related to the command type down by that user as he/she gradually enters the command. however , if the user enters the wrong word, the list is shrunk down to all the existing commands possible and the word itself is highlighted to mark the wrong spelling. And also the software will accurately and securely commit and retrieve data wherever they have been stored and its deliveries in compliance with the format expected.

3.5.2   Availability

Being a load and run software, VCT has to operate on any Operating System we can find on the market today (Windows, MacOs, Linux, and the oncoming Fuchsia).the software should be available in the cloud, the cloud being the various networked storage places determined by the client and easily accessible by a user to download in reasonable period of time.

3.5.3   Security

The aspect deals with any kind of Tektronix information disclosure which could be unethical regarding the company policy but also illegal, for the software should prevent anyone not entitled to get access to company data, by accidentally retrieving them from the secure storage location.

Furthermore, the VCT application data should securely be transmitted across the network and well stored in the Database.
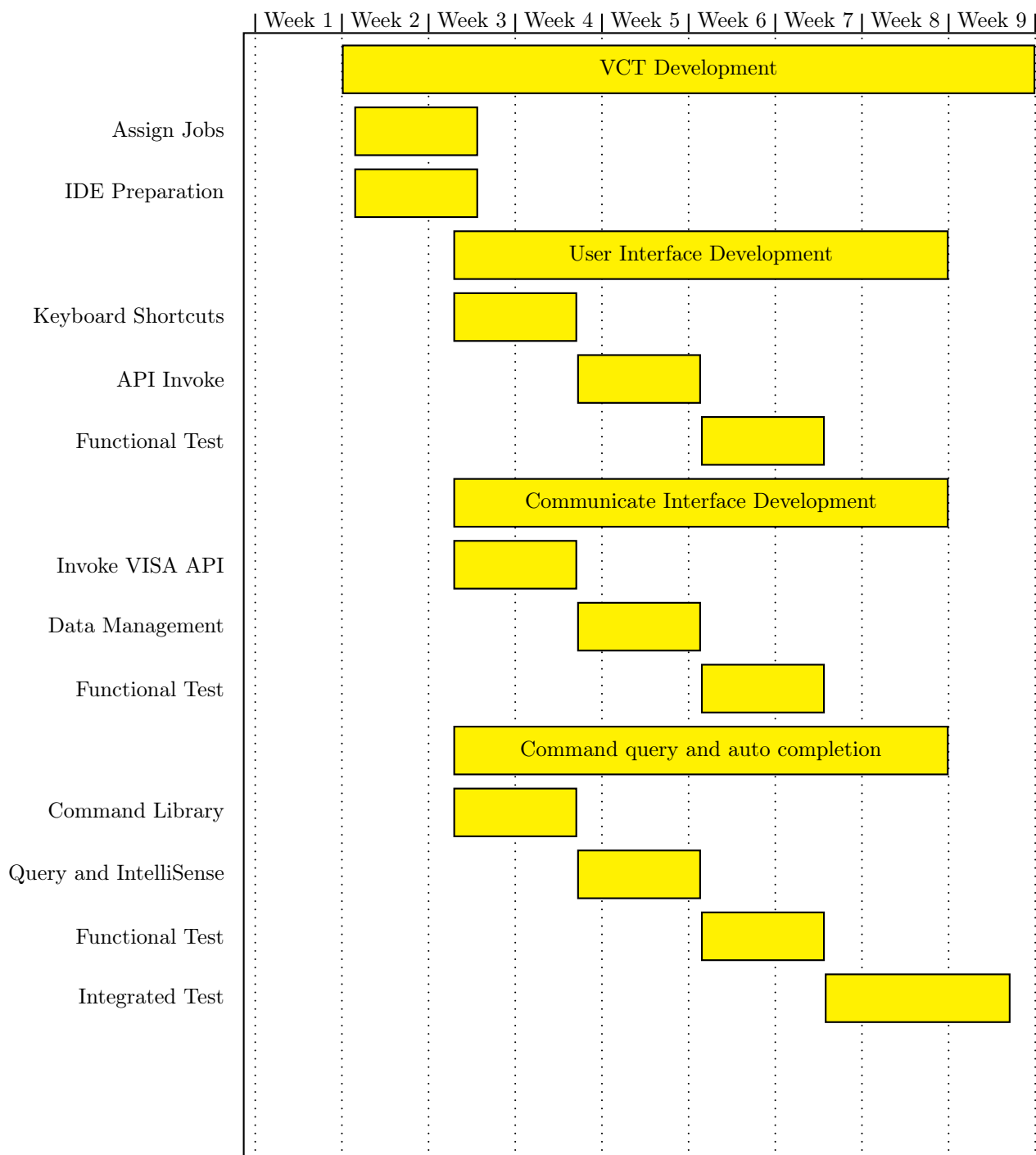
3.5.4 Maintainability

The maintainability being defined as the capability of the software to undergo some rearrangements of it code in order to yield better performance and even to integrate new modules and thus, to be easily updatable. Nonetheless, the software can allow a user to perform some light maintenance on the user interface.

3.5.5 Portability

Here comes into play, one of the key feature of VCT which is load and run, the means that the user on any operating system would be able to connect to the server on the accessible network download the application and runs it on the platform of his or her choice. In other words, the software has to guarantee its high capability of portability without any crash, miscommunication over the communication interface which, in fact, is transactional point where the VCT hands over the communication to the operating system by if possible encrypting raw data (at application layer of OSI model; OSI: open systems interconnection model).

4 Gantt Chart

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9

VCT Development

Assign Jobs

IDE Preparation

User Interface Development

Keyboard Shortcuts

API Invoke

Functional Test

Communicate Interface Development

Invoke VISA API

Data Management

Functional Test

Command query and auto completion

Command Library

Query and IntelliSense

Functional Test

Integrated Test

Please Sign in this page: