ORIGINATOR:

Name	Job Title	Signature	Date
Brian Cullinan	Software Developer		5/18/12

APPROVALS:

Name	Job Title	Signature	Date
Scott Pletzer	Director of Quality & Program Management		18MAY12
Andrew Mason	Chief Technology Officer		5/18/12

REVISION HISTORY

Revision	Revision Date	Reason for Revision/Change Request	Revised By
000	18May12	Initial Release	S. Pletzer

1.0 Purpose

This document provides a description of the ClearView Shell used to limit the accessible functionality of Windows in order to deter viruses and misuse of the ClearView system.

2.0 Scope

This document describes at a high level the various software components that make up the ClearView Shell. The specific requirements and specifications for the ClearView Shell software are captured in the specific requirements and specifications located in the DHF and/or controlled through EPIC's quality system.

3.0 Definitions

Shell	Software used to access features of the operating system (Windows XP) such as viewing tasks (e.g. ClearView) and running and closing applications.

4.0 Operational Environment

The ClearView Shell is software that runs on the Windows XP/Windows 7 environment. It is displayed automatically when the user logs in to a non-administrator account.

5.0 Functional Description

The primary function of the ClearView Shell is to limit the access and ability to misuse the ClearView system:

- Limits which programs can be run.
- Provides minimalistic interface for launching ClearView.
- Provides interface for administratively limiting default Windows functionality.

5.1 Limits programs

The shell only allows 3 options, run ClearView, Logoff and Shutdown. All other programs are excluded.

5.2 Minimal interface

The shell only shows a start button, and a clock and a button for ClearView when it is launched. When run as an administrator there is a settings option.

5.3   Limit administration

The settings dialog provides options for limiting administration of the shell. Options include disallowing the shell to be closed, disallowing access to task manager, and setting the desktop background to the ClearView logo.

6.0   Development Environment

This section describes the development environment used to develop and support the Clinical Trial Web-form software.

6.1   Programming Languages and Tools

The Clinical Trial Web-form software has been developed using Microsoft .NET Framework Version 4 in a Microsoft Visual Studio 2010 Development Environment. The majority of the application has been written in managed code, C# being the language of choice. The Shell uses the Windows API to interact with launched programs.

6.2   Environment

The development environment consists of various development systems dedicated to specific functions. These systems are dedicated to user interface development, reporting development, computational development, and testing.

6.3   Configuration Management and Version Control

Configuration management and version control is handled through Perforce, a SCCS (Source Code Control System), which is specified in EPIC's Software Development procedure, EG-010. Software development lifecycle is specified in EPIC's Software Development Lifecycle procedure, EG-018.

6.4   Documentation

All software documentation is contained in several sources including:

6.4.1 Software Requirements Documents

The entire product life cycle is contained in the software requirements specification (SRS) documents and summarized into an Input/Output Matrix table. The software requirements specifications (denoted as SR-004 in the Design History File) contain all of the software requirements for the ClearView Shell software.

6.4.2 Detailed Software Design Specifications

Detailed software design specifications were developed to document the implemented functions of the software in response to the requirements documents. The detailed specifications (denoted as SS-003 in the Design History File) provide specific detail regarding the functionality available to users of the ClearView Shell software. The specifications serve as a basic for testing design and include screen shots of the implemented functionality as available to users.

6.4.3 In Code Documentation

An open source product called Sandcastle was used to support source code documentation. Sandcastle is a documentation compiler for managed class library that generates Microsoft-style Help topics, both conceptual and API reference. It creates API reference topics by combining the XML documentation comments embedded in our source code with the syntax and structure of the types which it acquires from reflecting against the associated .NET Framework assembly. All assemblies are documented at the Public visible scope.

7.0 References

Site Creation, IT-001
Software Development, EG-010
Software Development Lifecycle, EG-018