

Software Requirements Specification

for

Towson University Object Lab

Prepared by COSC 412.001

Spring 2015

Instructor: *Dr. Sandro Fouché*

Course: 412.001

Date: 3/25/2015

Contents

REVISIONS

1 INTRODUCTION

- 1.1 DOCUMENT PURPOSE
- 1.2 PRODUCT SCOPE
- 1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW
- 1.4 DEFINITIONS, ACRONYMS AND ABBREVIATIONS

2 OVERALL DESCRIPTION

- 2.1 PRODUCT PERSPECTIVE
- 2.2 PRODUCT FUNCTIONALITY
- 2.3 USERS AND CHARACTERISTICS
- 2.4 OPERATING ENVIRONMENT
- 2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS
- 2.6 USER DOCUMENTATION
- 2.7 ASSUMPTIONS AND DEPENDENCIES

3 FUNCTIONAL REQUIREMENTS

- 3.1 GENERAL
 - 3.1.1 PRINTERS
 - 3.1.2 FILES
 - 3.1.3 STUDENT USERS
 - 3.1.4 ADMIN USER
- 3.2 USER INTERFACE
 - 3.2.1 LOGIN
 - 3.2.2 DISPLAYS
- 3.3 FILE MANAGEMENT
 - 3.3.1 STORAGE
 - 3.3.2 FILE HIERARCHY

4 NON-FUNCTIONAL REQUIREMENTS

- 4.1 SAFETY AND SECURITY REQUIREMENTS
 - 4.1.1 SECURITY
- 4.2 SOFTWARE QUALITY ATTRIBUTES
 - 4.2.1 RELIABILITY
 - 4.2.2 MODULARITY
 - 4.2.3 CORRECTNESS
 - 4.2.4 USABILITY



Revisions

Version	Primary Author(s)	Description of Version	Date Completed
OLI SRS 2.0	COSC 412.001 Spring 2015	The software will enable students to submit files to a specified 3D printer or laser cutter. It will allow staff to approve or reject files, send students personalized emails explaining the reasoning behind rejected files, and accept user input for the specific details of the file.	03/25/15

1 Introduction

1.1 Document Purpose

The software requirements were collected from the client. The Object Lab Interface (OLI) is a modified and enhanced version of its predecessor TomSoft. OLI will be an aid for the Towson Object Lab that will handle the file tracking and migration of objects printed in the lab. Objects can be printed via a variety of 3D printers or by a laser cutter. OLI's central purpose is to better the Towson University Object Lab's work flow by increasing efficiency through a reliable process that records desired details of objects printed in the lab.

1.2 Product Scope

Software will increase efficiency in the object lab by reducing the manual steps needed to track information. It will achieve this by organizing files and keeping the desired printers' information through one program. The end result of the project will be a documented install process that will allow for easy software migration in the future.

1.3 Intended Audience and Document Overview

This document is intended for Professor Fouché and future COSC 412 classes. This SRS contains the software requirements and is organized by section and line. We intend for this document to help us organize the project requirements and better assess priorities. Additionally, it is meant to help future classes understand what the software does, was intended to do, and what problem it was meant to solve.

1.4 Definitions, Acronyms and Abbreviations

Admin - Teachers aid, teacher, or faculty that approves files to be printed, creates build, and views data.

Art Department Hardware Device - term used to reference any additional printers, cutters, or other hardware devices that may be added in the future. These devices need to have data generated from a process that has a need for tracking and would otherwise be done manually.

Basic Computer literacy - possessing sufficient familiarity with modern computing devices to be able to log into an operating system, operate a mouse and keyboard, run an executable, create a folder, and follow simple on screen instructions.

Build - one or more 3D printer files make up a build. The build software makes a build of objects that will be printed by the printer. These objects are manipulated inside and around each other to maximize printing efficacy. File types including: .stl .obj .zpr as of now these are the minimum file

types that must be accepted the addition of more is to be determined. Builds can be made for any devices defined by the Art Department and has an agreed value for the lab.

Client - teacher, administrator, or privileged user of the system; The end user of the software.

OLI (Object Lab Interface) - refers to the new software being developed by the Spring 2015 COSC 412-001 class, for the Towson Object Lab; formerly called TomSoft by the previous class and the names may be used interchangeably.

Students - Submit objects to be approved for printing.

TomSoft (Towson Object Management Software) - see OLI.

2 Overall Description

2.1 Product Perspective

OLI is a product that creates and keeps records of all files created within the Towson University Object Lab that are desired to be tracked. The student will submit a file to the admin and the admin will decide whether to approve it or not. The data including student information and details for the object will be recorded regardless of whether the file is approved or rejected. OLI does not interact with the printers only students and admins.

2.2 Product Functionality

The major functions of the system are to allow students to submit and store data about files submitted for review. There it will be approved or rejected by the Administrator. The projects will be stored in a storage location. One storage will be for accepted projects and the other for denied projects (Jess 3/4). The files are accessible from anywhere that the software is installed (Josh 2/16). OLI keeps a record of all the data that is generated from the build files, meaning it will keep a record of all materials and the time it took for each file to print along with who requested the print. That build information is entered in by an administrator.

2.3 Users and Characteristics

There will be two types of users that will access OLI. The user-types in order from most access to least access are listed below:

1. Admin - They will have administration level privileges and should be able to easily access and modify all of the information stored. The admin will use the software most frequently and will have permission to approve or deny submitted files.

2. Students - They will only be able to submit a file. The file types they can submit are specific to the Art Department Hardware Device they chose to submit to. Each individual user will be using the software infrequently so OLI should be simple and intuitive

2.4 Operating Environment

OLI will be compatible with computers running Windows Vista and Windows 7 either 32 or 64-bit with Java version 8 update 25 or greater. Also, OLI will support multiple user access across different computers. This entails that one user accessing the system from an individual computer will not interfere with a different student accessing the system from a separate computer at the same time.

2.5 Design and Implementation Constraints

The primary issue is the memory requirements for the storage of the projects. There are over 700 files of student submissions that are required to be stored.

1. The system must be dynamic to allow the adding and removing of printers.
2. Will run on the following 32-bit and 64-bit Windows systems:
 - a. Windows Vista
 - b. Windows 7

2.6 User Documentation

Create the following:

1. A step-by-step installation guide
2. A user manual guide for students
3. A user manual guide for admin

2.7 Assumptions and Dependencies

We are assuming that:

1. Currently, the printers the software will address are the Zcorp Zprinter 250, Object Desktop 3D, or the SolidScape R66t.
2. The Zprinter will only need to track the size of the model.
3. The Desktop 3D will need to track the weight of the model (g).
4. The SolidScape R66t will track volume and time, however not very accurately.
5. The laser cutter used will be the VLS 2.30 and it has a large variance in the time a project will take.

3 Functional Requirements

3.1 General

3.1.1 Printers

1. OLI can handle the files for the Zcorp Zprinter 250.
2. OLI can handle the files for the Objet Desktop 3D.
3. OLI can handle the files for the SolidScape R66t.
4. OLI can handle the files for the VLS 2.30 laser cutter.
5. OLI can handle new file types for new Art Department Hardware Devices.
6. Must be able to add new Art Department Hardware Devices.
7. Must be able to remove Art Department Hardware Devices.

3.1.2 Files

1. OLI should be able to obtain and store information about the projects including student name, student ID, project name, amount of material used, and file name.
2. OLI should be able to obtain and store the reason a file was rejected if the file is rejected.
3. The file location along with the appropriate information should be stored.
4. The file location should be able to be updated if the file is moved.
5. OLI should keep record of the rejected files for each printer.
6. OLI should keep a record of the files pending approval for each printer.
7. OLI should keep a record of the approved files for each printer.
8. Files should be able to be manually moved.
9. Files for the object lab printers are stored in a separate drive.

3.1.3 Student Users

1. Should be able to submit files for the object lab printers.
2. OLI should allow for multiple users to log in to the system from separate computers in order to submit files to the admin to be approved or rejected.

3.1.4 Admin User

1. Should be able to open the submitted files via the appropriate software the object lab already has and reject or approve the submission.
2. Create excel reports similar to the report provided by the client through a semi automated process.
3. Notify student users that their file was accepted or rejected.

3.2 User Interface

3.2.1 Login

1. Separate login for user and administrator.
2. Secure password and user-name based login for administrator.

3. Towson University ID number based login for users.
4. If first time user login, ask to create an account with a name and email address.

3.2.2 Displays

1. Easy to navigate menus that:
 - a. Allow users to return to previous screens
 - b. Include conspicuous log out functions
 - c. Include simple project submission button
2. Link to the "Help" file in the UI.
3. The GUI should always remain on top of the screen.
4. Able to open files from the GUI in the appropriate player application.

3.3 File Management

3.3.1 Storage

1. Create a separate online accessible drive to store the project files.
2. Files must be able to be moved manually by the user from outside OLI.
3. Notify when the drive is nearing its limit via email to the administrators.
4. Facilitate the submission of two files with the same name via file renaming.

3.3.2 File Hierarchy

1. Create a pending job folder, to be printed folder, completed folder, build folder and a reject folder for each of the printers.
2. The software should be able to archive the files to save for future reference.

4 Non-functional Requirements

4.1 Safety and Security Requirements

4.1.1 Security

1. Must enter a password in order to access the administrator role.
2. Password must not be displayed in plain text while it's being typed in.
3. Student must supply their Towson ID to submit a new project.
4. Administrator must approve a submission to print.

4.2 Software Quality Attributes

4.2.1 Reliability

1. Must be able to reliably transfer files around in storage, with no loss or corruption to a percentage of 100%.
2. Changes to student-submitted files must not crash OLI

4.2.2 Modularity

1. Must be able to alter presentation of data based on which printer the data is associated with.

4.2.3 Correctness

1. Must report precise measurements of volume to an accuracy of 100%.
2. Must report material used to an accuracy of 100%.
3. Must report amount of time spent printing to an accuracy of 100%.

4.2.4 Usability

1. Installation will be executable from a package that will require minimal effort from the client.