UnBox3D:
Software Requirement Specification

Version
4.0.0

Group 1: Alexander Ramirez, Vivian Cases, Jacky Lim,
Nicholas Sisneros, Vince Wang

Contents

Version Description

Author	Date	Description	Version
Vivian Casas	December 11, 2025	Update for snapshot 1	1.0
Vivian Casas	December 11, 2025	Update for snapshot 2	2.0
Alexander Ramirez	December 11, 2025	Update for snapshot 3	3.0
Alexander Ramirez	December 11, 2025	Update for snapshot 4	4.0

1 Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the UnBox3D application. This is the final snapshot of the project. Here, our goal is to add the option to unfold the model and add any last touches to the user interface. To do this, we will update the user interface to add the Unfold button, implement the unfolding operation, and add any last touches to finalize our work. This document outlines the system architecture, core components, user interfaces, and design considerations that support the conversion of complex 3D models into simplified geometric representations and corresponding 2D layouts for fabrication. This document serves as a foundation for understanding the requirement specifications and implementation strategies behind UnBox3D.

1.2 Intended Audience

This document is intended for software developers, project managers, and stakeholders involved in the design, development, and deployment of UnBox3D. It may also be of interest to engineers and defense personnel who will use the application to streamline prototyping and fabrication processes.

1.3 Overview of the Software

The UnBox3D application is a software tool that bridges the gap between digital modeling and physical assembly. UnBox3D allows a user to import a model in .obj format and have it displayed in a special viewport where the user is given the option to simplify the imported model and, once satisfied with their simplifications, export the model to 2D cutouts for fabrication.

1.4 User Interface

- **Splash Screen:** The logo is shown as the program loads on the monitor

- **Import:** The import button is now functional, allowing users to either drop a .obj file into the viewport or click to open a file dialog to select the model they wish to import.

- **Rendering Screen:** A viewport where the imported 3D object can be viewed and rendered using OpenGL.

- **Object Hierarchy:** A list of all the objects within the imported 3D model. Here, we can select the object we want to simplify.

- **Simplify:** A button that triggers the simplification process on the currently selected object.

- **Simplify Slider:** A slider that allows removing objects from the scene if they are smaller than the size threshold specified.

- **Export:** A button that triggers the export process to convert the user's simplified model into 2D cutouts for fabrication.

1.5   Software Interfaces

- OpenGL is used to render imported 3D models.

- Assimp will be used in order to parse the data from the imported models.

- OpenTK will be used as a wrapper so that we can work with OpenGL in C#.

- Blender Python API is used to unfold the model.

2   Legal and Ethical Considerations

2.1   Data Storage and Privacy

At the moment, UnBox3D does not collect or store user data, other than temporarily reading the files that are imported. If persistent data storage is introduced, the application will comply with relevant data protection regulations such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA). Users will be informed of any data collection and given options to manage their privacy preferences.

2.2   Legal/Ethical Issues

Unbox3D is a general-purpose tool for converting 3D models into 2D unfolded layouts for fabrication. Although the initial sponsor is a U.S. Army research organization and one intended use is the generation of mock training targets, the software is not limited to military applications. It can also support civilian, educational, and commercial use cases such as architectural models, prototyping, hobbyist projects, and training aids. The development team therefore strives to keep the software functionally neutral and broadly applicable, minimizing direct coupling to any single military workflow.

From an ethical standpoint, contributors should be aware that any tool capable of improving fabrication efficiency in a military context may indirectly support armed conflict. At the same time, the tool can enable safer training, reduce material waste, and provide value to non-military users. The project is considered ethically acceptable under the assumption that it remains general-purpose, does not incorporate offensive or targeting logic, and is

documented in a way that encourages peaceful and educational applications in addition to the sponsor's needs.

Legally, Unbox3D is required to be built entirely on free and open-source software (FOSS) components. The final project license is expected to be a permissive or copyleft FOSS license (e.g., GPLv3 or a BSD-style license), ensuring that recipients retain the right to inspect, modify, and redistribute the source code in accordance with the chosen license terms. All third-party libraries and tools (including Blender and OpenTK) must be used in compliance with their respective licenses, and no proprietary or paid dependencies are to be introduced.

Users are responsible for ensuring that any 3D models processed with Unbox3D respect intellectual property rights, export-control regulations, and organizational classification rules. The project does not grant any rights to redistribute imported models; it only provides a mechanism to transform models that the user is already authorized to use.

3   Glossary

Acronym	Definition
GDPR	General Data Protection Regulation
CCPA	California Consumer Privacy Act
Assimp	Open Asset Import Library
OpenTK	Open Toolkit Library