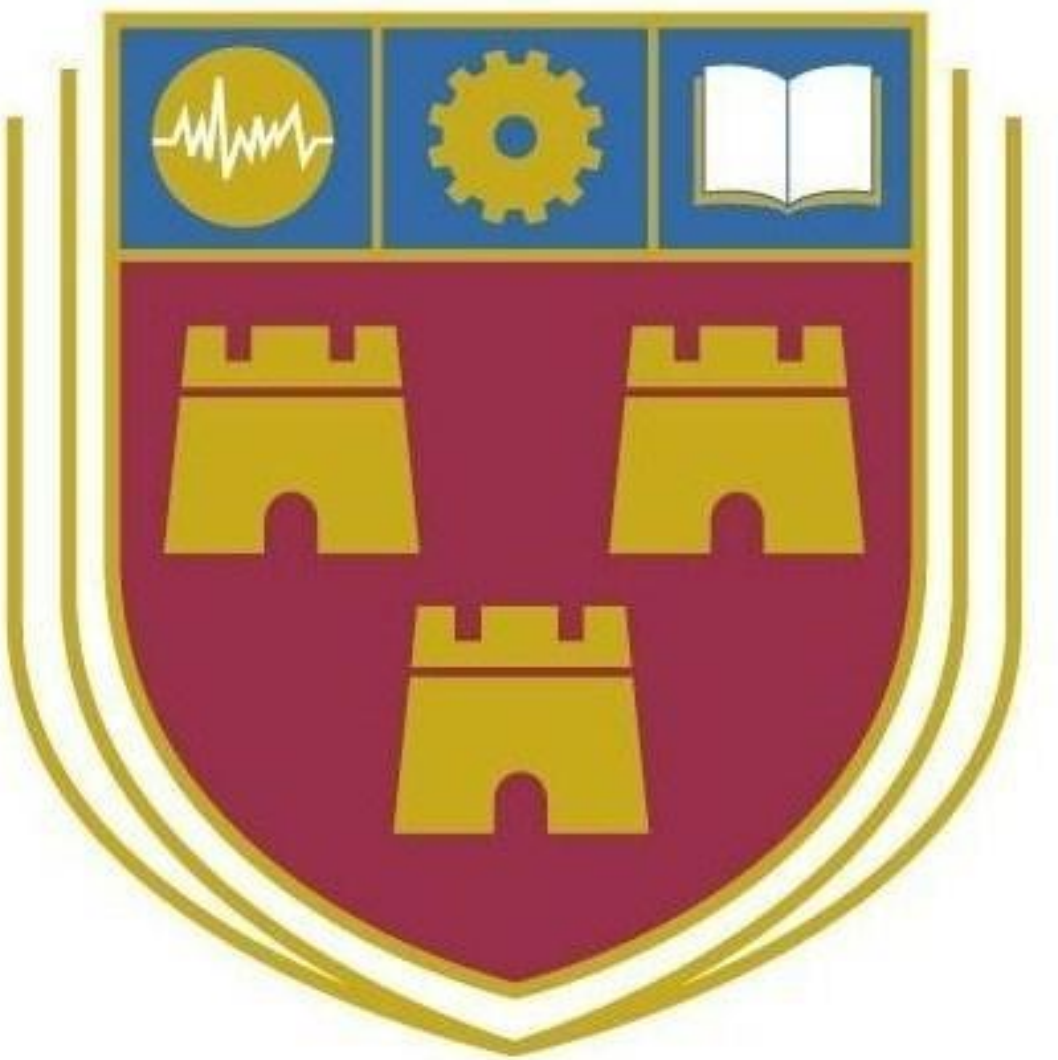
Step File Converter Functional Specification

Alan Halpin - Institute of Technology Carlow

Started 16/11/2020



Supervisor: Joseph Kehoe
Student Number: C00229361

Table of Contents

Abstract

The purpose of this project is to create a tool that aids with CNC printing. The target users being industry engineers. The tool aims to automate the process of vibration analysis on a CAD piece. This tool should be capable of reducing cost and time spent by engineers manually performing vibration analysis. This process usually takes an engineer several weeks to complete, this tool aims to complete that process in a single day. This tool is a proof of concept.

Introduction / Project Description

The STEP File Converter is a tool for CNC engineers, designed to automate a lengthy and costly process. Nothing like this application currently exists. The value is derived from the process being automated. This process involves performing vibration analysis on different permutations of an object until a satisfactory sequence of faces is found. Vibration analysis is performed to ensure the piece worked on remains stable throughout the cutting process. The proposed tool will extract geometrical data from a STEP file and catalogue the different faces of the object described. One by one these faces will be subtracted from a solid block shape. Vibration analysis is performed between each step, if the piece does not pass the requirements then a different face will be used. This will eventually create a list of faces to cut in order. Ex. A,B,F,G,C might be an order in which to carve first. Although vibration analysis already exists within various CAD applications, the process described only exists in a manual implementation. This tool will use the Solid Works API to outsource the calculations and identify the optimal cutting order within the application. An engineer will pass a STEP file to the program and the program will start working on the results. Depending on the parts complexity this process may take some time.

Context Diagram

An engineer passes a STEP file to the client. The client extracts the geometrical data and catalogues the different features. The client then sends features one by one to the solid works simulation API. The API performs vibration analysis and returns the result. If this result is satisfactory then another feature will be sent. This process is repeated until the optimal cutting order is found. If / when a feature does not pass the requirements then a new feature is sent instead(new permutation generated).

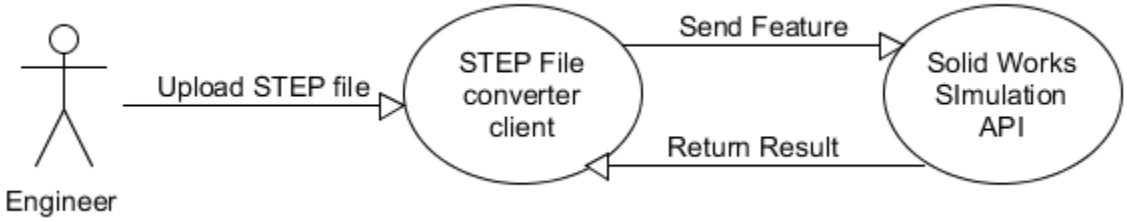


Figure 1: System Context Diagram

System Use Case Diagram

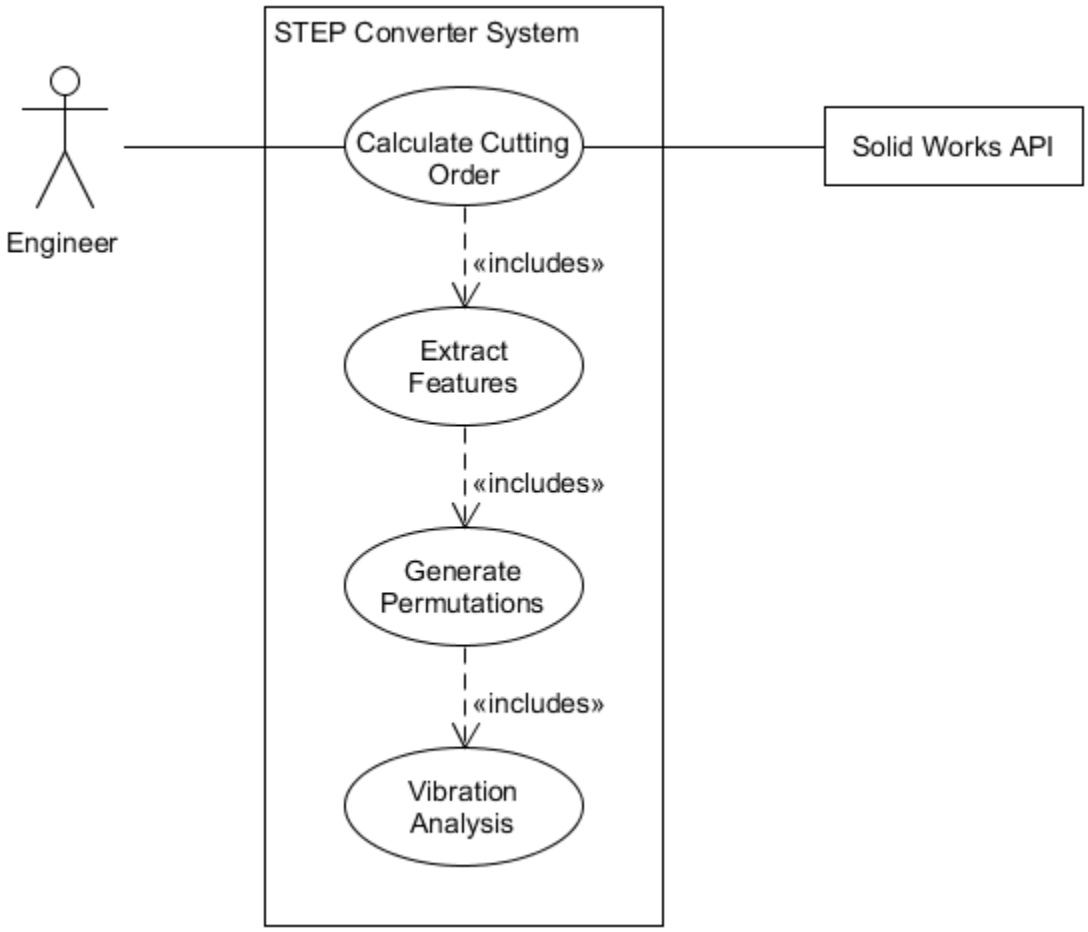


Figure 2: System Use Case Diagram

Use Cases

As this use case is complex, we will break it down into several smaller use cases.

Name:	**Extract Features**
Actors:	Engineer (User)
Description	Use case for extracting geometrical features from STEP file.
Preconditions	A valid STEP file has been uploaded.
Main success scenario:	1. The user tells the system to begin extracting features. 2. The STEP file is read into memory and catalogued. 3. A recursive algorithm is used to find advanced faces and their sub-properties. 4. Advanced face is found. 5. Sub-property(ies) is found and stored locally. 6. All properties have been located, and the system is informed of success.
Post Conditions	The features have successfully been stored.
Alternative	4a. Advanced face is not found. 1. System is informed of failure. 2. Application is restarted. 5a. No sub properties found. 1. System searches for the next advanced face.

Name:	**Generate Permutation**
Actors:	NULL
Description	Use case for generating permutations of potential cutting order. The lexicographic order algorithm is used for permutations.
Preconditions	Features have been stored locally
Main success scenario:	
Post Conditions	Permutation is produced.

Name:	**Vibration Analysis**
Actors:	Solid Works API
Description	This use case involves exporting faces one by one to the solid works API in order to perform vibration analysis. The system will pass a feature and wait for a result.
Preconditions	A permutation has been generated.
Main success scenario:	1. Feature is passed to Solid Works. 2. Feature is subtracted from a solid block piece. 3. Vibration analysis is performed using Solid Works. 4. Results are sent back to the system. 5. Results are received. 6. Results pass, next feature is passed. 7. All features have been passed and results have all passed. 8. A cutting order has been solved.
Post Conditions	The user is displayed the cutting order.
Alternative	1a. No features available 1. System is informed of failure. 2. Application is restarted. 6a. Results do not pass requirements 1. New permutation is requested from permutation class. 2. New permutation is received. 3. Next feature is passed.

Application (FURPS+)

FURPS+ is a model used to define functional and non-functional requirements of software. This technique helps prioritise more important functionality while also defining less important functionality. The acronym stands for: Functionality, Usability, Reliability, Performance, Supportability and the '+' stands for non-functional requirements such as design, interface, implementation or constraints.

Functionality

The main features of the application and the functionality.

Main Features:

1. Extract Geometrical Data from STEP
    a. The ability to read in a STEP file and extract various features and faces.
2. Catalogue / Store Features
    a. The different faces and features will need to be stored as they will need to be accessed again and again throughout the process.
3. Export Features to Solid Works API
    a. The faces and features will need to be exported to Solid Works API in order to calculate the vibration analysis for each step of the process.
4. Generate Permutations
    a. If a feature does not meet the requirements when the results are returned from solid works, a new permutation will be generated. An algorithm that generates permutations is required.
5. Display Cutting Order
    a. Display the order of faces in which to cut the piece.

Secondary Features:

1. User Interface
    a. The ability to use the application via UI.

Usability

Usability refers to the user's experience with the application, specifically the user interface. Usability states requirements such as accessibility, interface aesthetics and consistency within the UI. The application should be accessible and traversable on a windows desktop. The user should be able to upload a STEP file within 30 seconds 99% of the time. The user should be able to find the optimal cutting order within 30 minutes 80% of the time.

Reliability

Reliability depends upon availability, accuracy and recoverability. The application should load without failure 99% of the time. The application should work without internet access 100% of the time. It should be 99% accurate in producing results. Failures should be recoverable 90% of the time.

Performance

Performance refers to system response time, recovery time, start-up time, retrieval of results. As this application will be working with the Solid Works API, a lot of these depend on the performance of Solid Works itself and the machine being used. Solid Works is a 3d CAD tool that is notoriously tasking on hardware. The transfer of data (Faces / Features) between the application and the solid works API should take up to 15 seconds 90 % of the time. Features / faces should never be permanently saved in the application or anywhere else.

Supportability

Supportability is determined by testability, maintainability, installability. The application should be compatible with any windows machine. The code should be sufficiently tested and contain plenty of comments to allow sustainability and maintainability in the future.

‘+’

Metrics

How the success of this project will be gauged:

- Application should run on windows
- Users should be able to upload STEP files to application
- Application should be able to read a STEP file.
- Application should temporarily store features from STEP file
- Application should successfully send and receive data to / from solid works simulation API
- The application should be able to determine the best order in which to carve out the features

Testing

How this application should be tested:

- Check if the application runs on windows.
- Check if a valid STEP file can be passed to the application
- Check if an invalid STEP file can be passed to the application.
- Check if the application can load a sample STEP file into memory and catalogue it.
- Check if the application is storing the faces and features correctly.
- Check if the application is receiving the correct response from Solid Works.
- Check if the produced cutting order is the same as a known cutting order.

Plagiarism Declaration

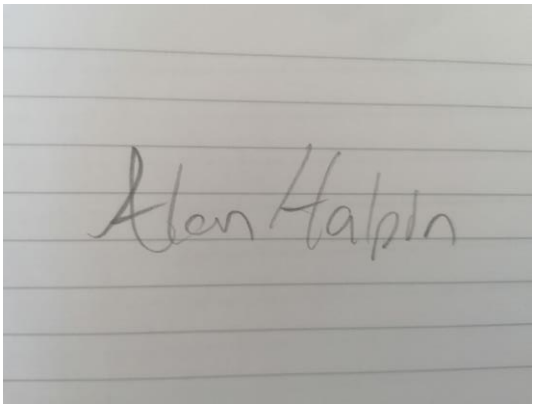
Declaration

- I declare that all material in this submission e.g.thesis/essay/project/assignment is entirely my/our own work except where duly acknowledged.
- I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside.
- I have provided a complete bibliography of all works and sources used in the preparation of this submission.
- I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offense.

Student Name: Alan Halpin

Student Number: C00229361

Signature:



Date: 30/04/2021