Report on Steel Portal Frame Design Project

Introduction

Project Overview

This project aims to design a steel portal fram using Python, leveraging the PySide6 framework for the GUI and OCC (OpenCASCADE) for 3D modeling. The steel portal frame consists of I-sections for the columns and rafters, with a layout of purlins on top. The frame's dimensions and properties are adjustable via a graphical user interface, which takes user inputs and generates the corresponding 3D model.

Project Overview

- To create a 3D model of a steel portal frame that can be visualized and saved to a STEP file
- To provide an interactive UI where user can input the dimensions of the frame
- To generate 3D model of the steel frame and display it
- To allow the user to save the generated model to a STEP file for further analysis.

Tools and Technologies

- **Python**: The core programming language for the project
- **PySide6**:Python library, used to build GUI for the user input and interaction
- **OCC(OpenCASCADE)**: Python library, used for 3D modeling and visualization of the steel frame.
- **STEP file forma**t: For exporting 3D model to a standard file format

Code Structure

main.py

This file serves as the entry point for the application. It initialize the application, creates GUI, and connects the user input to the model generation function.

geometry.py

This file is responsible for creating the steel portal frame ,it contains functions that generate i-section, layout of the purlins and at the end assembles the portal frame in 3D space.

gui.py

This file defines the user interface. It includes a series of input fields(QLineEdit) for the user to specify the dimensions and other parameters of the frame components.

file_operations.py

This file contains the function to save the 3D model to a STEP file.

environment.yml

This file contains all the necessary dependencies require to run this project.

Prerequisites

- Conda installation
- Python specifically 3.6 to 3.10 version

After these run below command into the bash:

- conda create myenv python=3.x
- conda activate myenv
- conda env update --file environment.yml --prune

Future Work

- **Comparison**: The project could be extended to compare two different steel frame portal, their cost and all the necessary field work.
- **Optimization**: The design could be optimized for manufacturing processes or material usage.

Conclusion

This project demonstrates how Python, PySide6, and OCC can be used to create a parametric steel portal frame design system. This framework can serve as a foundation for more complex engineering design applications.