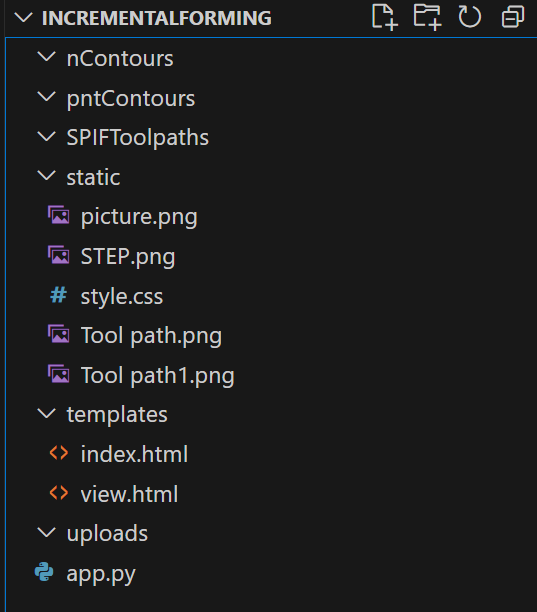
1. Structure:

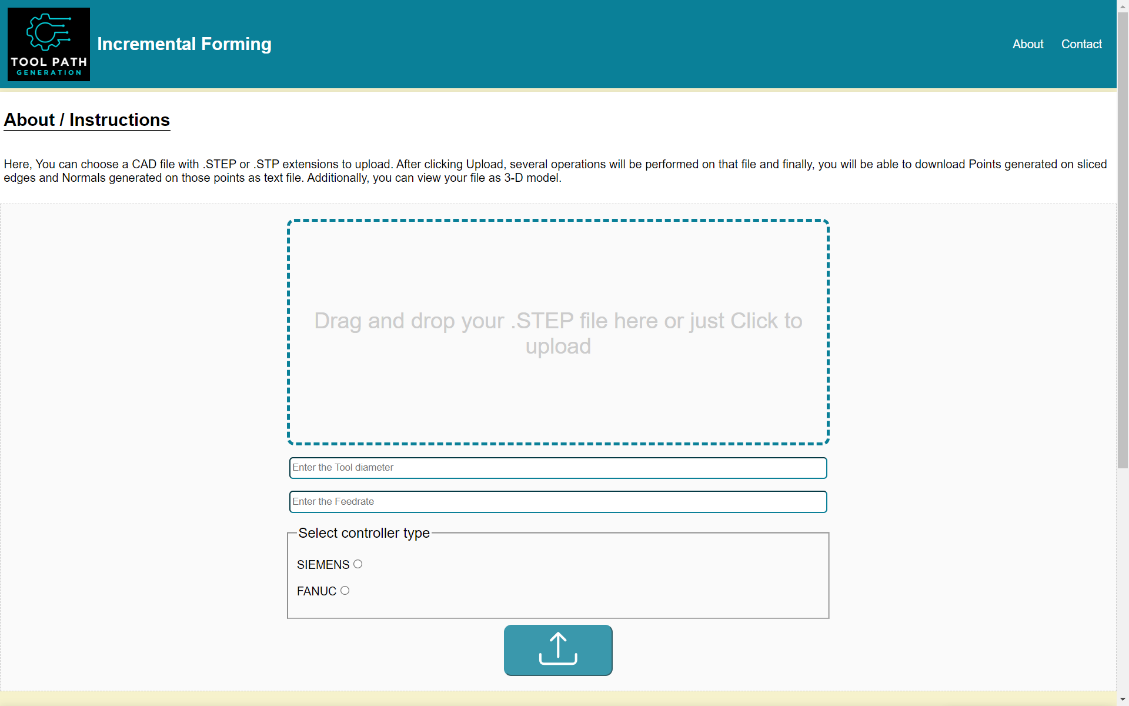


There are few folders namely, ***templates, static, uploads, nContours, pntContours, SPIFToolpaths*** and a python file *app.py*. Inside ***templates***, two html files, *index.html and view.html* are there. Folder ***static*** is used to store all the necessary things that we are using in our website like images, backgrounds, logos, icons, documents, etc. Also, it stores CSS files that is *style.css* in our case. Folder ***uploads*** is used to store .STEP file uploaded by user and .STL files created by the program. Folders ***nContours, pntContours, SPIFToolpaths*** stores normal, points and toolpaths respectively, that has been generated by the program. All these are inside single folder ***IncermentalForming.***

1. Working:

On running the python program i.e. *app.py*, which is a **flask**-based appusing the syntax *python3 app.py* an URL will be generated using which we can access our website in any browser.

Below is the landing page of our website i.e. *index.html.*



This upload button will submit the form

To select desire controller

Here, you can upload any .STEP file.

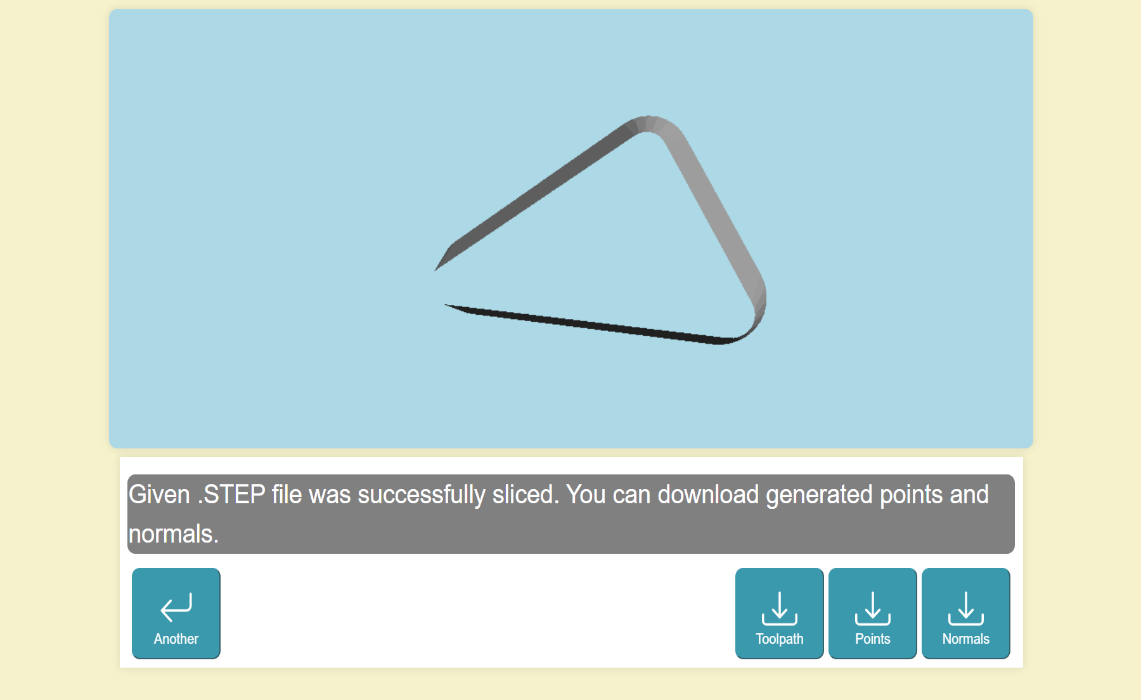
To enter required parameters

After clicking upload button, several background operations will be performed on the uploaded file, which includes conversion to .STL in order to display, slicing, generation of points on sliced edges, generation of normal on those points, generation of toolpath for incremental forming machine using those points and normals. These operations are being performed using a python library called *Python OCC*, which is already imported in *app.py.*

After all that, a new page will open i.e. *view.html*, which will display the CAD model and will have download buttons to let user download points (.txt file), normals (.txt file) and toolpath (.mpf file).

Return back to home page

Download Buttons



Display

Return back to home

Download Buttons

Clicking any of the download buttons will trigger a function in *app.py* that will get the fille stored in any of ***nContours, pntContours*** or ***SPIFToolpaths*** to get downloaded in user’s system.

The display canvas is displaying the CAD model uploaded by user using a JavaScript called *three.js* library. Here a function in *app.py* first converts the .STEP file to .STL and the it’s been displayed using *three.js.*