**Team Name**: saSBOL Extension
**Team Members:** Stephen Sweet, Jin San Yoon, Antonio Alonzo, Nikolay Popov

We are the saSBOL extension team. Our tool is an extension to SBOLCanvas which given an input SBOL2 file of a genetic design, returns a sequence accurate SBOL2 file with the proper scars assembled. The user chooses from a selection of assembly standards and our tool returns their original design with implemented scars. This returned SBOL2 file can be uploaded back to SBOLCanvas and further edited.

Our ***readXML.py*** and ***XMLfunc.py*** scripts are responsible for reading the input SBOL2 (XML) files and extracting key information for our implementation.

Our ***Scar_Addition.py*** function is responsible for generating the necessary scar sequences given the users input and assembly method choice. It then returns the scar information to our ***XMLfunc.py*** script which creates the output SBOL2 (XML) file for the user. The communication and implementation of these previous scripts is done our ***main.py***

Our ***views.py*** is the only file the user must run in order to use our tool. This script launches the local host site to operate saSBOL. All other files are part of the front-end implementation/dev-ops efforts. Our web app is compatible with Docker and the Docker registry, as well as Heroku cloud services. However, while in the Heroku HTTPS cloud server we are unable to access client's local paths as per security constraints. Which is why it is currently on a local host/docker image. In the future, we have the ability to expand the functionality to the entire web.

Due to the current construction of our tool, it is important the user clones/forks our repository and DOES NOT change any of the paths, folders, directories of any files. To ensure proper functionality, user must simply run ***views.py*** and maintain the same structure of repository. Results will always be written to the DOWNLOADS_FOLDER and user can obtain them from there.