

**Processing and Visualizing Land Use Model Scenarios**

INFM 737 – Information Management Capstone Experience

**Client:** National Center for Smart Growth and Research 

**Developer:** Paridhi Mathur

**Project Coordinator:** Dr. Kathy Weaver

**Abstract:**

The National Center for Smart Growth Research and Education launched PRESTO, an advanced suite of integrated models, to develop and explore alternative futures for the Baltimore-Washington region. One of the models in the suite is the Simple Integrated Land Use Orchestrator (SILO) model which is used to simulate household relocations, demographic changes, and developments which add, upgrade, or demolish dwellings. It uses data from the following sources:

* The American Household Survey to create individual households and their dwellings.
* The Maryland Statewide Transportation Model (MSTM) for travel times of individuals by auto and public transit.
* Data from exogenous forecasts for projected employment locations.

Along with these inputs, other demographic indicators such as birth rate, marriage rate, death rate, household relocation rate are provided externally using national averages. The model micro-simulates a synthetic population to generate the output files. These output files describe the synthetic population of households, dwellings, persons, and jobs for the years 2000, 2015, 2030, and 2040. Additionally, SILO also generates two summary files: one with detailed data for the entire study area (such as persons by age and gender, number of dwellings by type, average commute distance, etc.), and one with selected zonal data (such as households or dwellings by zone, zonal accessibility, etc.).

The researchers in the PRESTO project use the output files to understand the key driving forces that would shape the future of the state of Maryland. They need to perform comparative analysis of the output files but currently lack a rapid and systematic approach. They require a stand-alone application to compare and analyze the various summary files generated by the Simple Integrated Land Use Orchestrator (SILO) model through a comparison table and visualizations. The objective of this project is to give them a tool to speed up the analysis and provide consistency in the comparison study of each simulation run.

**Project Overview:**

A stand-alone application needs to be created that enables the researchers using the SILO model to compare these output files with the help of a comparison table and to generate visualizations of the different indicators present in the files. The main objective of the comparison of is to help researchers gain insights about the key driving forces behind the various land use scenarios for the state of Maryland. The following highlights the tasks for this project:

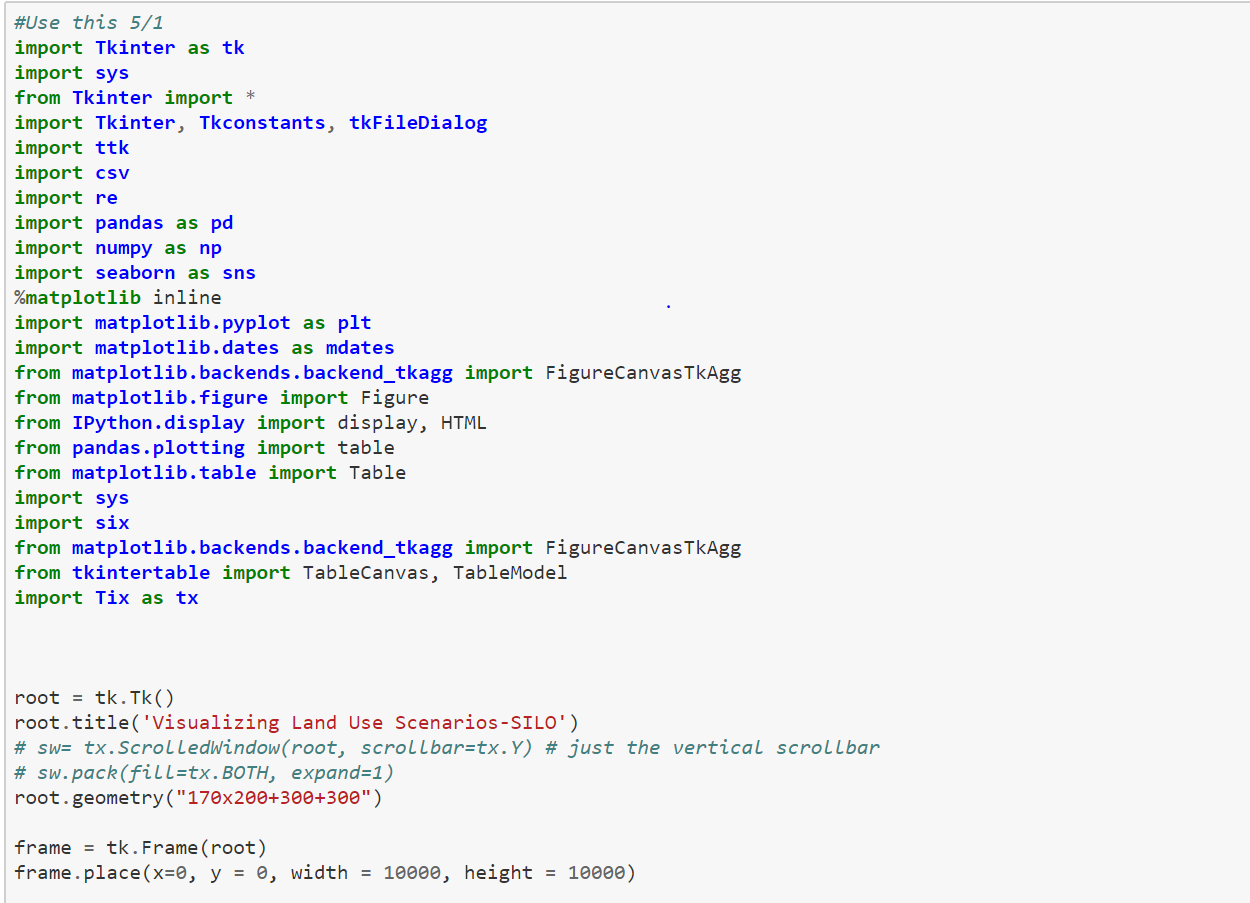
1. Research about the programming language to be used for development.
2. Gather detailed information about the output files.
3. Develop the comparative analysis module.
4. Develop the visualization module
5. Test the complete application.
6. Deploy the application.

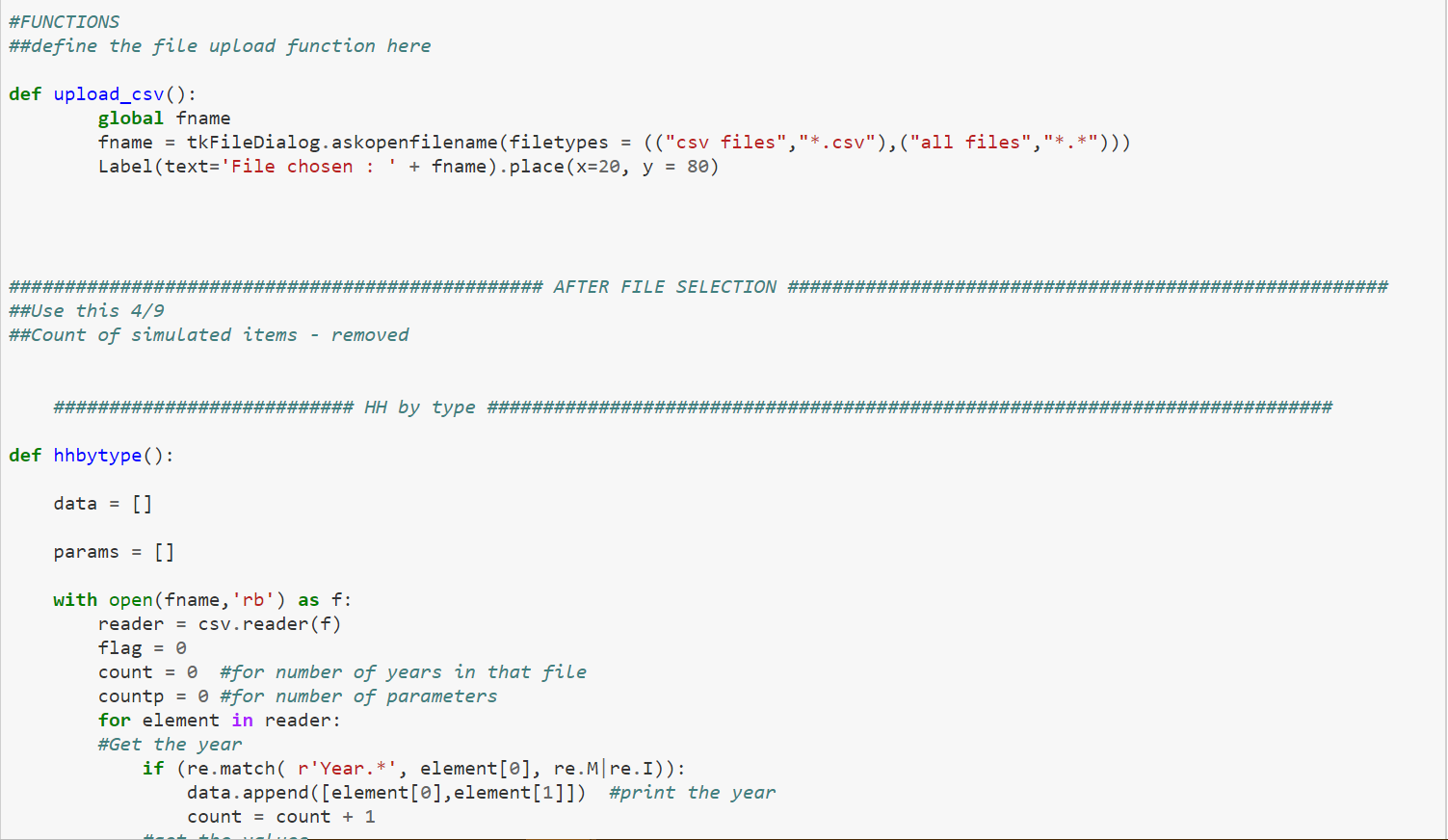
The final deliverable of the project is a stand-alone application that would perform the comparative analysis of the output files and create visualizations of indicators. The primary stakeholder of this project is the PRESTO team which includes researchers, students, and interns. All the stakeholders would use the application to achieve a single goal, so they would be referred collectively throughout the rest of this document.

**Project Implementation:**

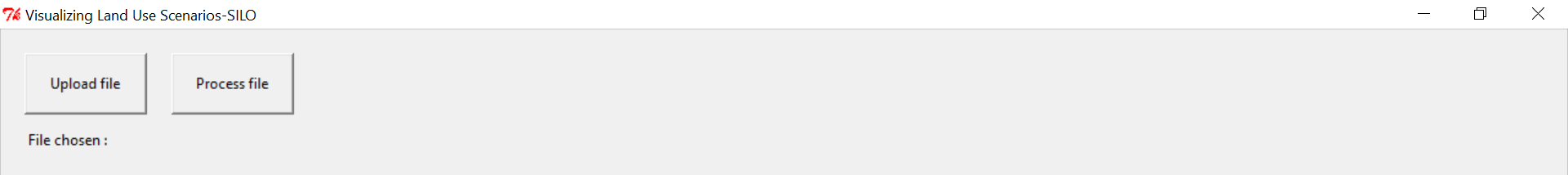
After researching the various possible development platforms, Python was selected for the several in-built libraries that support application development. Python additionally eliminated the need for having a database backend. The files could be accessed through a dialog box functionality in Python. Described below is a step-by-step functioning of the project along with a part of the Python code.

Python code snippet:



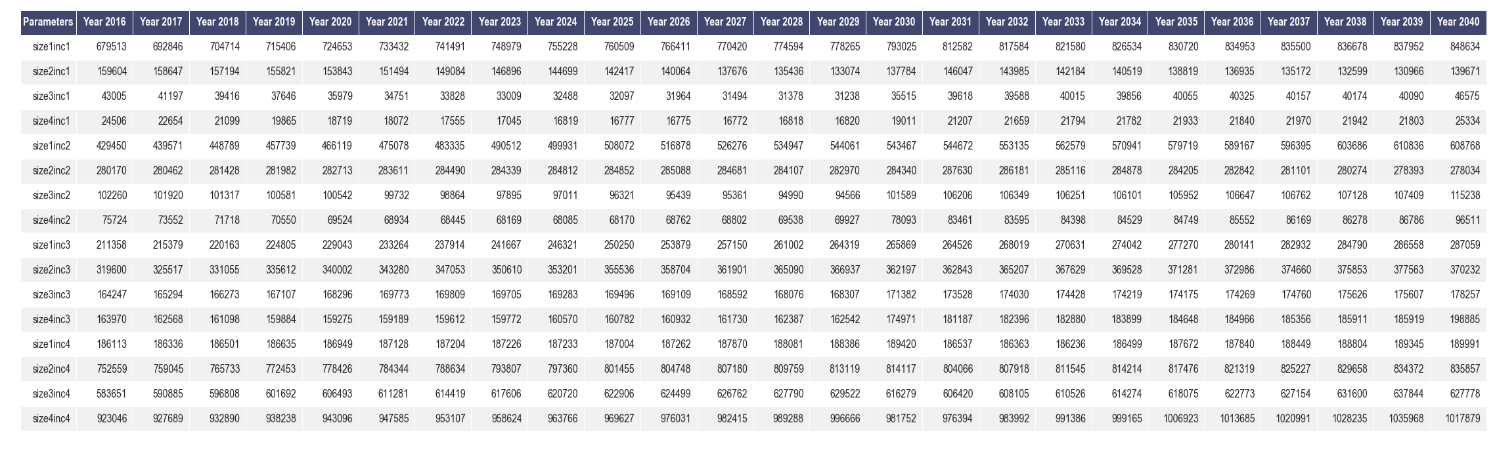


After running the above file, the below window appears:



The user then uploads the required file and clicks on ‘Process file’ button. The results are then displayed on the window.



****

**Successes and Challenges:**

One of the biggest challenges of the project was the identification of the best platform for the application development. The client did not favor any particular development language and the decision was based on project requirements and ease of implementation. In the requirements gathering phase, it was decided that the application would be built using Java but later the decision shifted to using Python as it was better suited for the requirements.

After moving to Python, the next challenge was to understand the unstructured csv file from which the data had to be extracted. The csv file was consolidated with all the metrics in different data structures and did not have any commonality in terms of storage. The code had to be optimized in such a way that the results were displayed in a tabular form and the application did not take too much time to execute.

The project was filled with personal successes as well. The process of simultaneously learning, understanding, and implementing Python presented very steep learning curve. While my background in computer science and data analytics helped in the basic understanding, a lot of credit goes to my colleagues who cleared the doubts along the way.

From a personal standpoint, delivering a project from the requirement gathering phase to the final client delivery was a learning experience. The experience not only taught me the importance of being accountable but also gave me the independence to plan and work on a project. It improved my understanding of the entire project development life cycle, the various deliverables, the timeline that a project needs to follow and also the importance of communication between the different project stakeholders.