**Important Features:**

1. **Clear:** The visualizer is designed in a way such that the user can easily tell what’s happening on the screen. The user must be able to keep seeing the big picture of the combinatorics problem at hand and not be distracted by any pop-up screens or drop-down menu’s.
2. **Easy:** As the target user does not have a lot of experience with combinatorics problems, the interface should be easy to understand and use. Even without fully understanding combinatorics problems, the user should be able to input and solve problems to get started on the topic.
3. **Visual:** As many features as possible should be implemented in a visual way. This includes inputting and editing a certain problem, or changing its parameters.
4. **Adaptable:** It should be easy for the user to change parameters for a given combinatorics problems and to see these changes have effect on the solution.

**Usage:**

First, the user inputs the structure of the combinatorics problem. The X–set (on the left) can be edited by adding elements or subsets (domains). For the Y–set (on the right) only the size can be edited. Constraints can be added when needed. When the problem setting is complete, the user can run the program. All possible arrangements will be iterated (step by step / animated).

