**Software Requirements Specification** (SRS) Document

**Project Title:** Probability-Based Object Distribution Simulator

**1. Introduction**  
 **1.1 Purpose**

This project aims to help users understand how objects can be distributed into drawers, providing insights into probability concepts and their real-world applications, such as logistics and resource allocation.

**1.2 Scope**  
 Simulate various probability-based distribution scenarios and provide visual representations to aid understanding

**2. System Overview**  
 The system will take input parameters, such as the number of objects and drawers, and determine the likelihood of different distribution outcomes based on probability rules.

**3. Difficult-to-Explain Areas**

**3.1 Probability Distribution Mechanics**

Probability distributions form the mathematical backbone of this project. Users may struggle with concepts such as combinatorial analysis (Understanding permutations and combinations in object allocation). To clarify these areas, interactive examples, step-by-step breakdowns, and visual will be incorporated into the project.

**3.2 Visual Representation Challenges**  
 Presenting probability data in an easy-to-understand format is crucial. The project must determine the most effective way to visualize distributions

**4. Tools and Technologies**  
 Programming Language: C#  
 Development Environment: Visual Studio

**Changelog #1:**

* Researched combinations of sets for object allocation
* Developed a class to generate combination sets
* Linked a WinForm UI to the class for testing

**Changelog #2 :**

* Improved random numbers generation for better results
* Used backtracking to generate combinations more effectively
* Made the UI update textboxes dynamically
* Researched how to interact with multiple Winforms

**Changelog #3 :**

* Implemented dynamic control creation for better visualization of distribution.
* Improved random combination generation by optimizing the backtracking algorithm.
* Refined WinForms interactions to improve consistency across multiple forms

**Changelog #4 :**

- Implemented dynamic control creation for visualization in non constrained distribution ( drawer can now be empty )

**Changelog #5: User Feedback**

- Implemented the code for non contrained distribution solution class

- made some research based on the user feedback: how to implement animation or use pictureboxes ( or panels? ) to represent the drawers instead of bars