

# Assumptions

## Lab Task 2: Sentence Generator App

Submitted By  
S.M.Mehrabul Islam  
SH-086  
22 August 2022

---

### 1. Design Pattern

After reading the problem description, it seemed suitable to use Strategy Design Pattern for this solution.

We have divided the functionality of the application into 2 distinct behaviors.

- a. Word Addition Behavior
- b. Generate Sentence Behavior

For each of these behaviors, we've implemented a separate strategy pattern using abstract class. The class diagram should explain the overall design of that.

So, in the future, if a new strategy is added for either of these behaviors, the solution should easily be extended without affecting existing strategies.

But if a new behavior gets added in the future, we might need to add another strategy pattern for that specific behavior.

### 2. Initial Dataset

Before the user adds any word to the sentence generator's dictionary, there must be some initial word dictionary.

So, we've used the list of the top 100 mostly used words in the English Language for initiating 3 separate word dictionary for 3 different sentence generators. Like below,

WordForRSG[100]	=	[ "the", "at", ... , ... , ... , "people", "part" ]
WordForSSG[100]	=	[ "the", "at", ... , ... , ... , "people", "part" ]
WordForOSG[100]	=	[ "EHT", "TA", ... , ... , ... , "ELPOEP", "TRAP" ]

In future, we will load a much bigger number initial words from CSV files which will contain the vocabulary for each sentence generators.

### **3. Volatile Data**

Currently, we are operating the functionalities on a runtime List/Array Data Structure instead of using something non-volatile like Files/Database.

So, any words that have been added to the sentence generator on runtime, will get lost if the application restarts.

But very soon, we're planning to implement the loading and storing of words using files, so that the application can maintain non-volatile data.

### **4. Variable Assumptions**

Some assumptions in the case of variables are:

- a. Minimum word length = 1
- b. Maximum word length = 18
- c. Word can not be empty
- d. Word can contain numbers or special characters, as we haven't yet added any filtering system for words other than duplicate detection.
- e. Currently, there are only 2 menu states & 4 possible menu options.

### **5. Feature Assumptions**

Currently, we are assuming that the user won't be able to delete any word from the word dictionaries. So, there's no Delete Word feature on the application yet.

And also, the user can't generate all possible sentence generation. They must generate separately if they want to use different generating strategies.