**Software Design Specification**

**(C# Application)**

by Jesus Aguilar-Andrade

**Programing 101Table of Contents**

[1. Introduction 3](#_Toc126747110)

[1.1. Document Outline 3](#_Toc126747111)

[1.2. Document Description 4](#_Toc126747112)

[1.2.1. Introduction 4](#_Toc126747113)

[1.2.2. System Overview 4](#_Toc126747114)

[2. Design Considerations 4](#_Toc126747115)

[2.1. Assumptions and Dependencies 5](#_Toc126747116)

[2.2. General Constraints 5](#_Toc126747117)

[2.3. Goals and Guidelines 6](#_Toc126747118)

[2.4. Development Methods 6](#_Toc126747119)

[3. Architectural Strategies 6](#_Toc126747120)

[4. System Architecture 7](#_Toc126747121)

[4.1. Subsystem Architecture 7](#_Toc126747122)

[5. Policies and Tactics 8](#_Toc126747123)

[6. Detailed System Design 8](#_Toc126747124)

[6.1. Classification 9](#_Toc126747125)

[6.2. Definition 9](#_Toc126747126)

[6.3. Responsibilities 9](#_Toc126747127)

[6.4. Constraints 9](#_Toc126747128)

[6.5. Composition 9](#_Toc126747129)

[6.6. Uses/Interactions 9](#_Toc126747130)

[6.7. Resources 9](#_Toc126747131)

[6.8. Processing 9](#_Toc126747132)

[6.9. Interface/Exports 10](#_Toc126747133)

[6.10. Detailed Subsystem Design 10](#_Toc126747134)

[7. Glossary 10](#_Toc126747135)

[8. Bibliography 10](#_Toc126747136)

# Introduction

This document was made to outline the proposed system design for the Coke Freestyle Machine. By integrating this new design to improve the way customers interact to get their desired drink and overall get business to prosper, this includes product tracking, improve maintenance, easy user interface, and customer satisfaction.

The purpose of creating such a System Design Document is to provide a description of construction on the new Coke Freestyle Machine. The Software Design Document was created to meet the Stake Holders specifications, which this document provides a description of how the specifications are met with software, and database design.

The completed software design should meet the following specification criteria*:*

* It should be able to re-order syrup when needed as well as carbonated water, so the team members and customers don’t have to worry about ever running out.
* The User interface should be easy to use and have an appealing appearance.
* It should be able to give reports to the business when the machine is low on product and statistics on the flavors use.
* It should be able to dispense the appropriate amount of liquid for the cup sizes that are available.
* The Machine should be able to dispense a mix of up to 3 flavors if the user decides to.
* The User interface should give you a breakdown of the product that will be dispensed

## Document Outline

* Introduction
* System Overview
* Design Considerations
  + Assumptions and Dependencies
  + General Constraints
  + Goals and Guidelines
  + Development Methods
* Architectural Strategies
  + Modular System
* System Architecture
  + Private subs
* Policies and Tactics
* Detailed System Design
  + Classification
  + Definition
  + Responsibilities
  + Constraints
  + Composition
  + Uses
  + Resources
  + Processing
  + Interface
  + Detailed subsystem Design
* Glossary
* Bibliography

## Document Description

### Introduction

The purpose of creating such a System Design Document is to provide a description of construction on the new Coke Freestyle Machine. This document should be able to provide a detailed description of development to accomplish the criteria. The document is intended for the business using the Freestyle Machine. To get data on which drinks and mixes of them are popular. First This document gives an overview of the system then, over design considerations, architectural strategies, system architecture, policies and tactics, detail system design, a glossary, and a bibliography of this document.

### System Overview

The Freestyle Machine software will be able to provide a user interface that will allow consumers to select the drink they will like and be able to mix different of their favorite Provide drinks if they choose to. Before pouring the drink, they will also be able to choose the cup size that they want to dispense just the necessary amount of liquid to reduce overall waste. The owners of the machine will have access to features such as manual syrup and Co2 replacements, syrup levels, soda statistics, and all in a fast easy to use interface.

# Design Considerations

Issues that need to be addressed before attempting to create the Freestyle Machine software we need to consider the specifications of hardware that will be running the program, as well as the data base structure that will be used to capture the data, and the input controls that will be used for functionality.

## Assumptions and Dependencies

There is an assumption that needs to be made in order to develop the software. With some of this in mind we will not just be able to complete the necessary functionality but be ablet to work within the constraints necessary. Does assumptions are the following:

* The .EXE file should run 24/7
* The operating system will be Window 7 or higher
* The display will be able to display color in HD
* End user will be able to identify the purpose of the interface
* Hardware isn’t exposed to harsh environmental conditions
* Multicore processor will be used
* 4gb of ram
* 50gb of storage
* Local Data Report should be easy to read.

## General Constraints

Constraints that we need to consider while developing this application would be the following:

* 2 core processors
* 50gb of Storage
* 4gb of Ram
* Subject to spilled liquids
* Syrup buildup
* Color blind
* 2 clicks away from dispense
* End-user environment
* Availability of Syrup
* Availability of Co2
* Health Compliance
* Food Laws
* Data storing requirements
* Security requirements
* Instant loads
* Wi-Fi connection
* The local data report can only fit a certain amount of data in one table

## Goals and Guidelines

Goals and Guidelines that need to be followed to deliver the appropriate solution to the client are the following:

* Access Keys should be implemented
* Tab Index should be correctly done
* Interface should be self-explanatory
* Exit button should close application
* Mix Up to 3 Flavors
* Be profitable for business to use
* Manual Re-ordering (Syrup and Co2)
* Easy UI usage (Color and layout)
* Be able to show Syrup levels
* Dispense flavors in 8oz, 16oz, 24oz, and 32oz
* Location Management Report
* Fluid Level Report
* Maintenance Report
* Order Report
* Have a three-table data base

## Development Methods

The development method used for this software solution is very modular to improve coding time, readability, and be able to easily modify code when needed. To Achieve this modular design there is a collection of Subs that are created and can be called any time to do and operation, and many of this Subs have code that allows calculations to be made instead of having hard coded numbers to provide certain statistic.

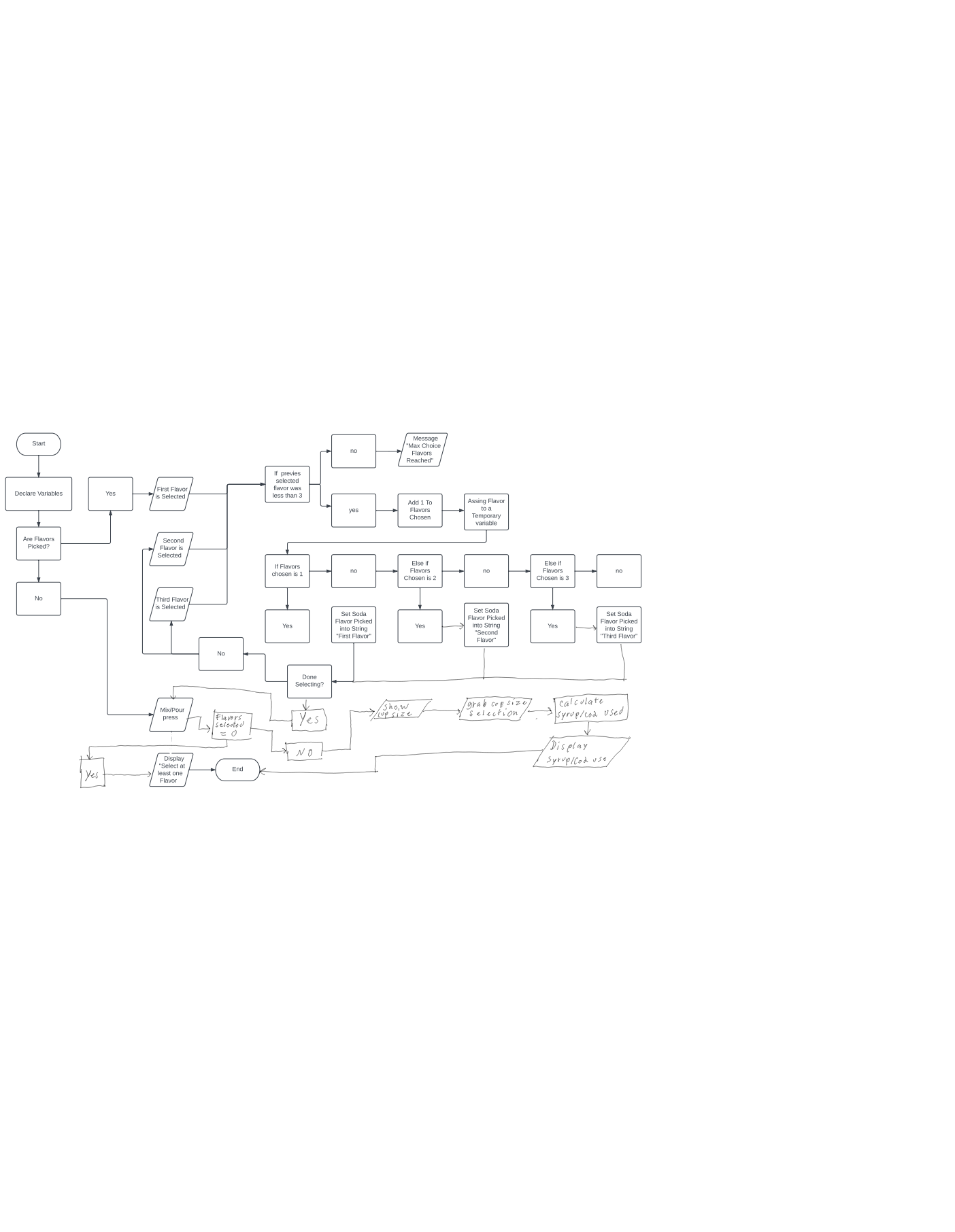
When it comes to the database the implemented tools in visual studio will be used, which connects the database with a dataset. The dataset is then used to save the necessary SQL statements.d

# Architectural Strategies

To design our program, we decided to use the Visual Studio Software by Microsoft as the IDE that it provides is a well stablish environment to program applications for windows, and the programming language that was developed by Microsoft C# .NET is perfect for this usage. The Icons used for the application were acquired from Coca cola in their already in use models to help people recognize the brands that they already love. In the future we plan to expand the number of flavors that will be available and improve the mix option to allow more than three flavors if the user chooses to. The user interface should be able to be used by a touch face input for easy use as well as accept Mouse input to facilitate development purposes. The program shouldn’t be hard to run so a multicore processor, 4gb or ram and 50gb of storage should be enough. When there is an error in the functionality, we should be able to have an error log that will help us keep track of these instances. An external data base will be used to store data of the machine usage and should be able to provide this data over internet at the owner's request. Since the storage requirement is low, we will want to upload the data after every dispense. Reorder of Syrup and Co2 are a core feature of the Machine as management can order the components from the machine itself and the new component should update the necessary fields in the data base such as expiration date, last fill date and update the amount left to full.

# System Architecture

The following is A flow chart that gives an overview of how the program works to give the desire outputs when inputs are made:



In the Flow chart above you may be able to see that some of the functions would require a lot of repetition but this is fixed using private subs to make Individual parts work together to provide the functionality. 4.1 will go over some of the private subs made in order to reduce programming time and redundancies.

## Subsystem Architecture

The private subs that were implemented are the following:

* Show and hide UI
  + hidegrpSize() -Hides the Group box
  + showgrpSize() -Show Group box
  + hidebtnMain() -Hide main Manu options
  + showbtnMain() -Show main Manu options
* MixCountCheck() -which Counts how many flavors were picked and limiting you to picking a max of 3 Flavors.
* SetFlavors() -which checks what flavor was picked in what order to set the name of the flavor to the appropriate strings which contain the first, second, and third flavor.
* DisplayStats() -which verifies that at least one flavor was picked and if there is in checks how many and it formats the info to be displayed as a string within a box that gives two options such as “ok” and “cancel”.
* CalcSyrupUsed() -This sub makes the calculations to dispense the necessary amount of Syrup and Co2.
* ShowNumPickFlavor() -It updates the display that shows how many flavors you picked.
* ResetSetFlavors() -Which resets the flavors that were picked.
* SubtractLiquidUsed() - Subtracts the amount of liquids used.
* ResourceLowCheck() - Checks if any of the liquids have met a threshold and displays a message box if it has.

The elements of the local data report are:

* Volume: Only managing the data that is coming from the Freestyle machine.
* Velocity: The data is updated after every order as long as the machine is able to upload such data to the database.
* Variety: All kinds of data on drinks in being collected such as profits, cost, orders, amount order, time, flavors used, popular flavors, popular mixes, number of orders, and time the fluids needed to be replaced or refiled.
* Veracity: Data will be reported in two tables, one that reports time specific data and another that displays day specific data.

# Policies and Tactics

* All C# .NET naming systems should be followed.
* A modular code design should be prioritized.
* Tab index should be properly implemented.
* Access keys should be present.

# Detailed System Design

## Classification

*N/A.*

## Definition

The Application is supposed to dispense the right amount of syrup and co2 that is required when a user wants to dispense their drink of choice.

## Responsibilities

*Jesus Aguilar = Programmer*

*Cody Rockwell = Programmer*

## Constraints

The constraints that we face are: The programming language, which is VB.Net, the knowledge that we have over the language, knowledge in database structure and time that we can use to work on the application.

## Composition

The Application is structured in a way such that the flavors are on the top of the application, so it is the first thing the user looks at. Then at the bottom of the interface is where all your functionality buttons are, such as the mix/dispense, statistics, and a group box that appears when the user wants to select the size of their drink.

## Uses/Interactions

A Costumer at a fast-food restaurant or restaurants team member would be the one interacting with the application.

## Resources

The **Inventory Table** Contains the following records:

* + **Primary Key**: Fluid ID – The Id correlates to the Position of the Container in the flavor Array (Co2 is Not contain in this Array).
  + Fluid Name – This Field contains the corresponding names of the Fluid ID.
  + Fluid Type – This Field categorizes the container whether it is a Flavor, Gas, etc..
  + Capacity (oz) – The information about the container capacity is recorded in this field.
  + Current Amount – This field is updated with the amount used to report the amount left of oz in the container.
  + Expiration Date - Here is where the expiration date of the container is labeled.
  + Last Fill Date– This field indicates the last time the product was refilled.

The **Order table** Contains the following records:

* + **Primary Key:** Order Id - A unique ID will be used to Identify an order.
  + Order Amount $ - This field contains how much the order was.
  + Date of Order – Contains the Date of the Order.

The **Order info table** Contains the following records:

**Primary Key**: This key is made up of two Foreign Keys (Order ID and Flavor ID)

* + **Foreign Key**: Order ID - A unique ID will be used to Identify an order.
  + **Foreign Key**: Flavor ID - The Id correlates to the Position of the Container in the flavor Array (Co2 is Not contain in this Array).
  + Is it mix - Indicates if the order had Mixed flavors or not.
  + Used oz – The amount of oz used out of the flavor container for the order.

## Processing

The application uses variables that are used by multiple private subs to add operation to an event when it happens, such as the Syrup icon click event, which changes the icon's appearance and stores the drink selected for future use.

## Interface/Exports

The application will export data and import data from the database from a remote location.

## Detailed Subsystem Design

*When the Syrup buttons are clicked the following happens:*

* + Count how many flavors are being use
  + Unique Flavor reference number is assigned.
  + Save the flavor to dispense by using the unique reference flavor num
  + Display the number of flavors that have being picked

When the Mix/Dispense button I s clicked the following happens:

* + Check if you have at least one flavor selected
  + If you do have a flavor or more selected, show the UI for cup sizes
  + Hide the main UI

When a cup size is selected:

* + Set the amount of soda to dispense
  + Make the calculation to dispense the right amount of syrup and co2
  + Display the amount of syrup and co2 was dispense
  + Deselect Option

# Glossary

**Freestyle**: Soda fountain created by Coca Cola that can provide a large amount of drinking options through only one nuzzle.

# Bibliography

N/A