

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

MODELACIÓN Y SIMULACIÓN 1

PROYECTO ÚNICO

PAREJA NO. 21

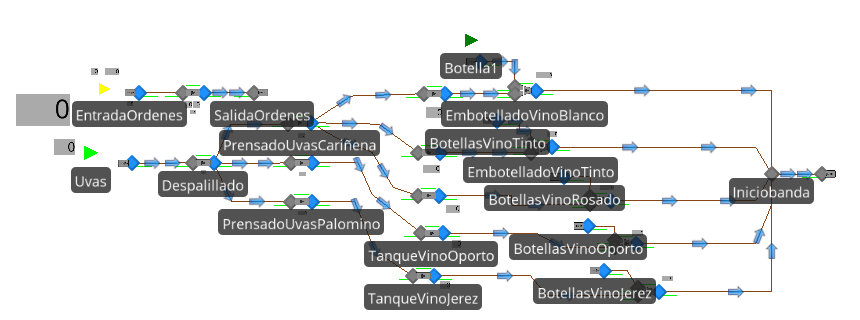
SEGUNDO SEMESTRE 2020

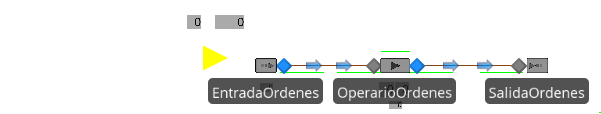
|  |  |
| --- | --- |
| Nombre | Carne |
| Aldair Estrada García | 201503855 |
| Edgar Rolando Herrera Rivas | 201520498 |

09/11/2020

# Elaboración de Vinos

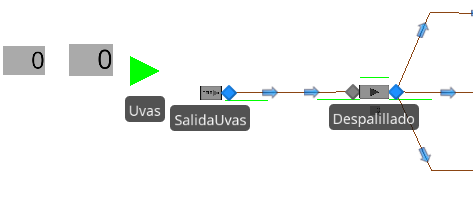
### Justificación del Modelo





Esta parte del modelo es donde se leerá el archivo Excel, se tomaran las ordenes para después ejecutar todo el proceso de elaboración de vino.

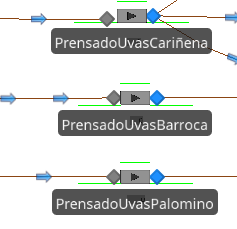
### Parte del Depalillado de Uvas



En esta parte se crea un proceso en el cual conforme se va tomado las ordenes de cuantas uvas necesita el cliente para hacer la cantidad de vino necesaria.

Luego para por una despalillada que consiste en quitarle toda la semillas o polillos que tenga la uva para dejar solo la uva limpia y lista para empezar la selección de vinos.

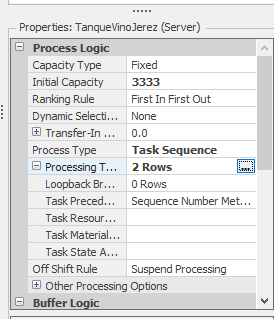
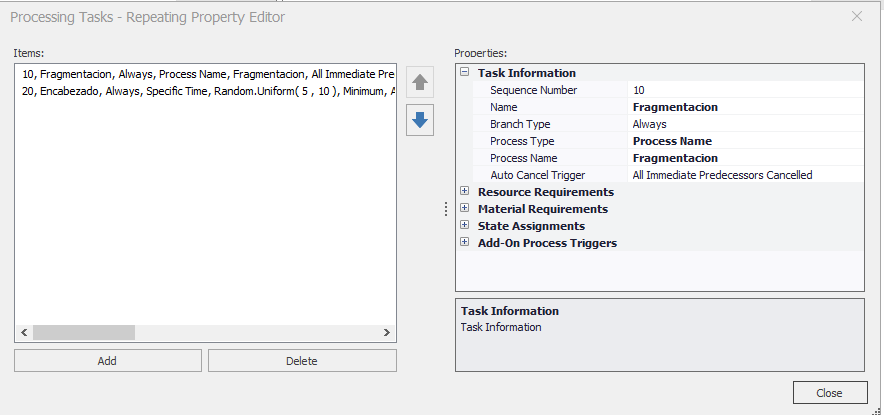
### Prensado de Uvas según tipo de vino



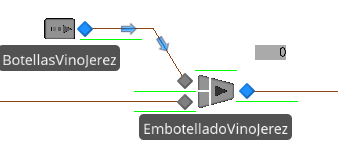
Proceso que consiste en deshacer la uva para sacarle el jugo.

## Tanques de Vino

Estos servidores lo que realizan el el proceso de fermentación y depende del tipo de vino realiza una segunda acción para el caso de Vino Blanco (Filtración) y para vino Jerez (Encabezado)



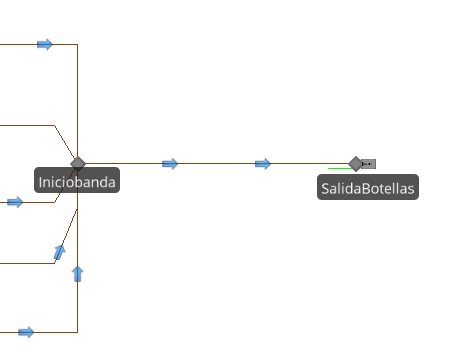
### Embotellado de Vinos



Este proceso es cuando ya pasa el proceso de fermentación y esta lista para llenar las botellas, Existen 4 tipos de Botellas (Estándar, Magnum, Doble Magnum, Imperial).

Procesos para todos los tipos de vino

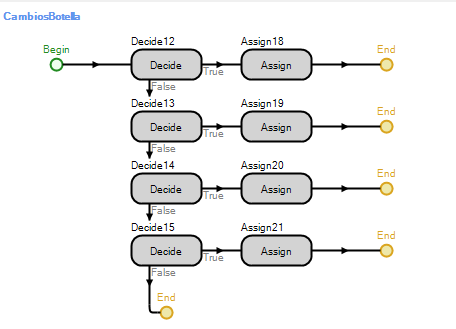
## Cola de botellas y Salida de Botellas



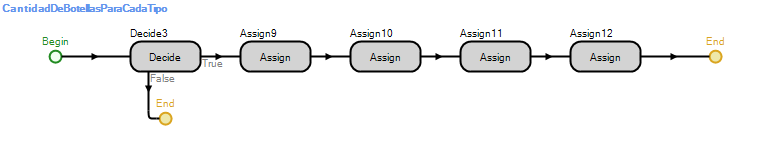
# Procesos

### Cambio de botella

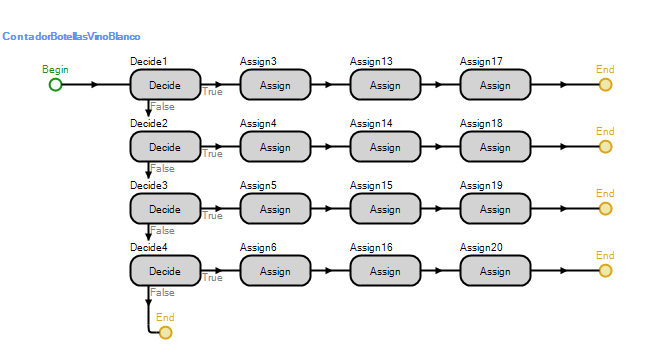
Proceso que realiza el cambio del tamaño de botellas



## Contador de Entidades de Uva para cada tipo de botella

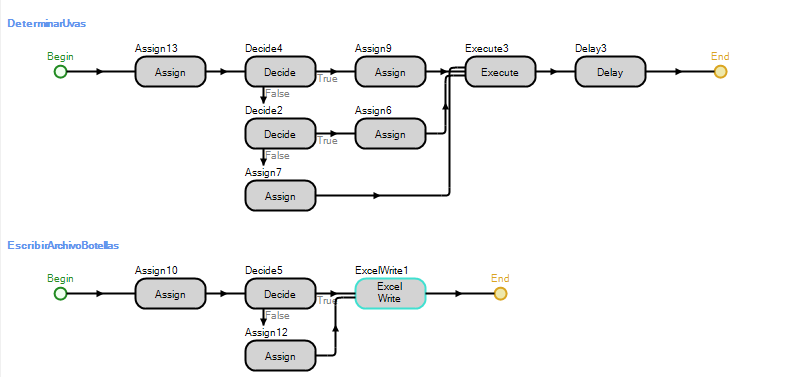


## Contador de Botellas según tipo de vino



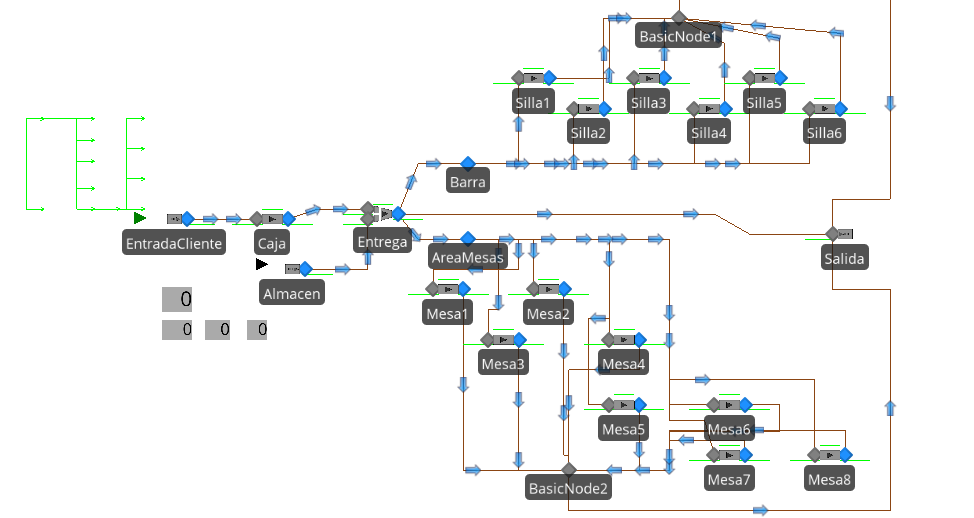
Proceso realizado para los 5 tipos de vino

### Determinar el tipo de Uvas y Escritura de archivo Excel con el reporte

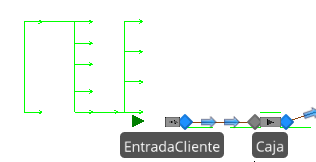


# Tienda 5

## Justificación del Modelo

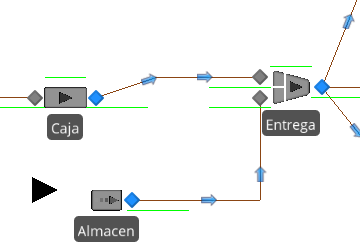


### Entrada del Cliente y caja



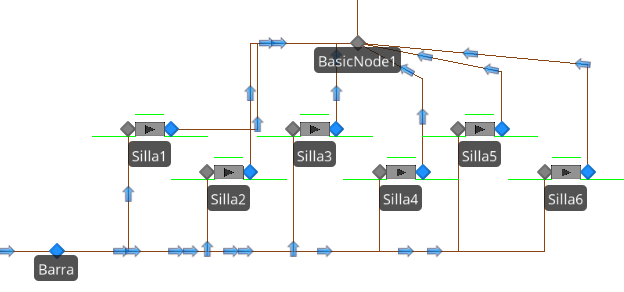
Se crea un Source y un servidor en el cual realiza la lectura del archivo de Excel

### Almacén, Caja y Entrega



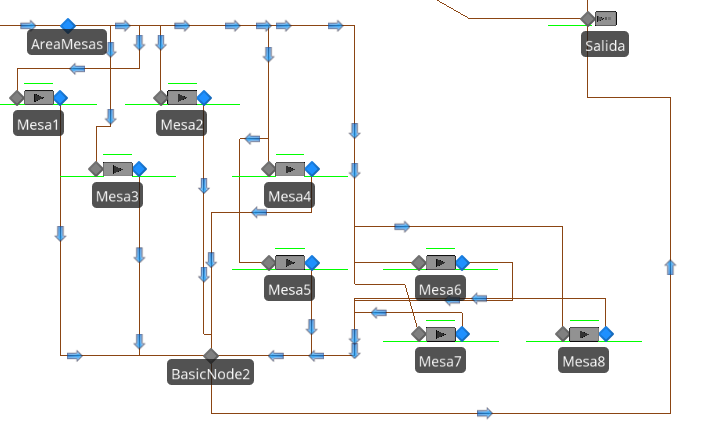
Aquí en donde el operario de la caja realiza el pedido al almacén para que le lleven la cantidad de vinos que solicita.

### Área de Espera Barra



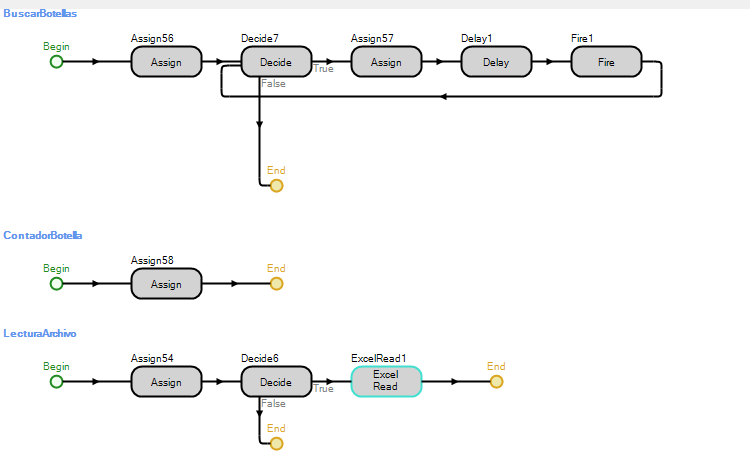
En esta parte del modelo el cliente puede ir a esperar a las sillas en lo que le llevan su pedido.

### Área de Espera Mesa



El cliente puede ir a esperar a las mesas en lo que le llevan su pedido completo

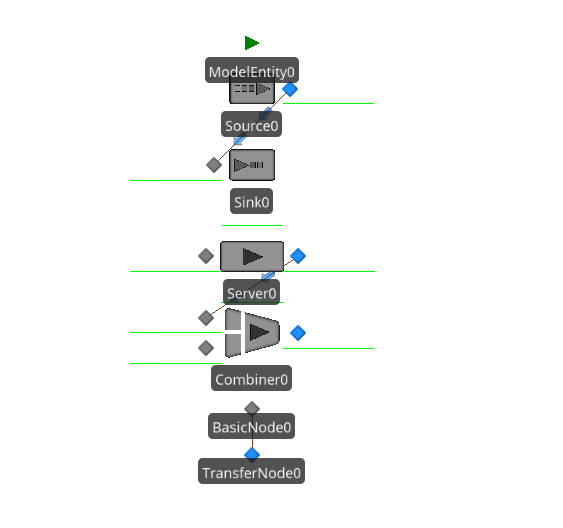
## Procesos



Procesos en los cuales se realiza la lectura del archivo, contador de botellas y el buscador

# API (Tienda 2)

# Justificación del Modelo Base



Para el modelo base es necesario colocar el cada uno de los componentes que se utilizaran en al API para el desarrollo del modelo final.

Componentes para utilizar

* ModelEntity : Este componente representa la entidad del modelo.
* Source: Es la salida de las entidades en un determinado tiempo
* Sink: Es el componente que destruye las entidades cuando ya cumplen su función en el modelo
* Server: Este componente se utilizará para demostrar la entidades que realizan una acción en concreto
* TransferNode: Estos componentes representaran puntos en los cuales se estará distribuiyendo el modelo con el fin de estar un poco mas ordenado.
* Path y TimePath: Representa las conexiones entre entidades ya sea por una distancia en concreto o por un tiempo especifico

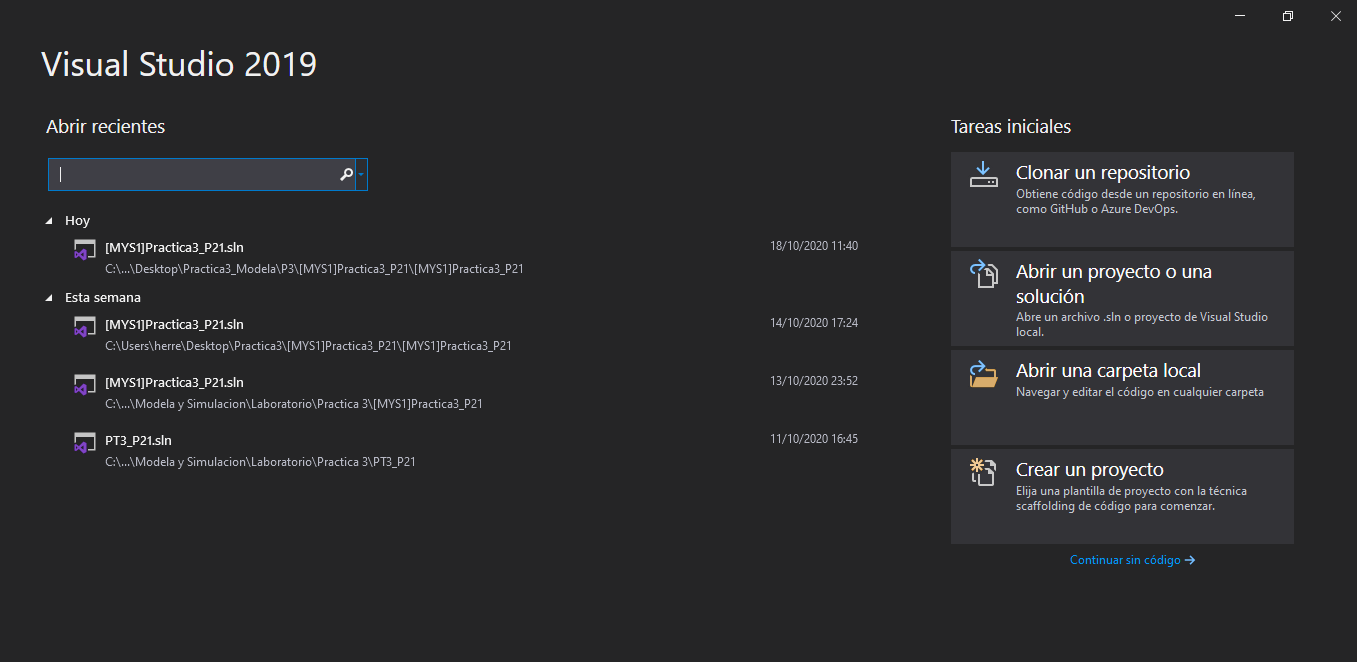
# Pasos para crear una Api en visual studio (C#) con simio.

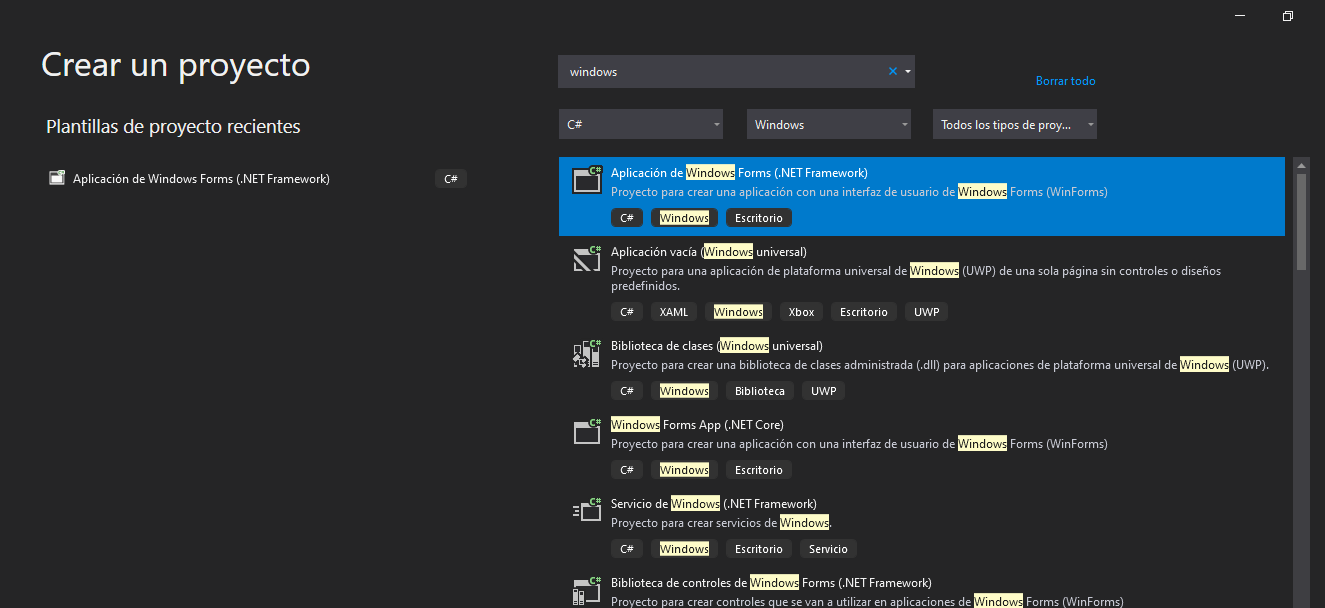
## Paso 1.

En este paso es necesario tener instalado visual estudio de preferencia versión 2019 ya que funciona mejor el programa de simio en este editor.

## Paso 2.

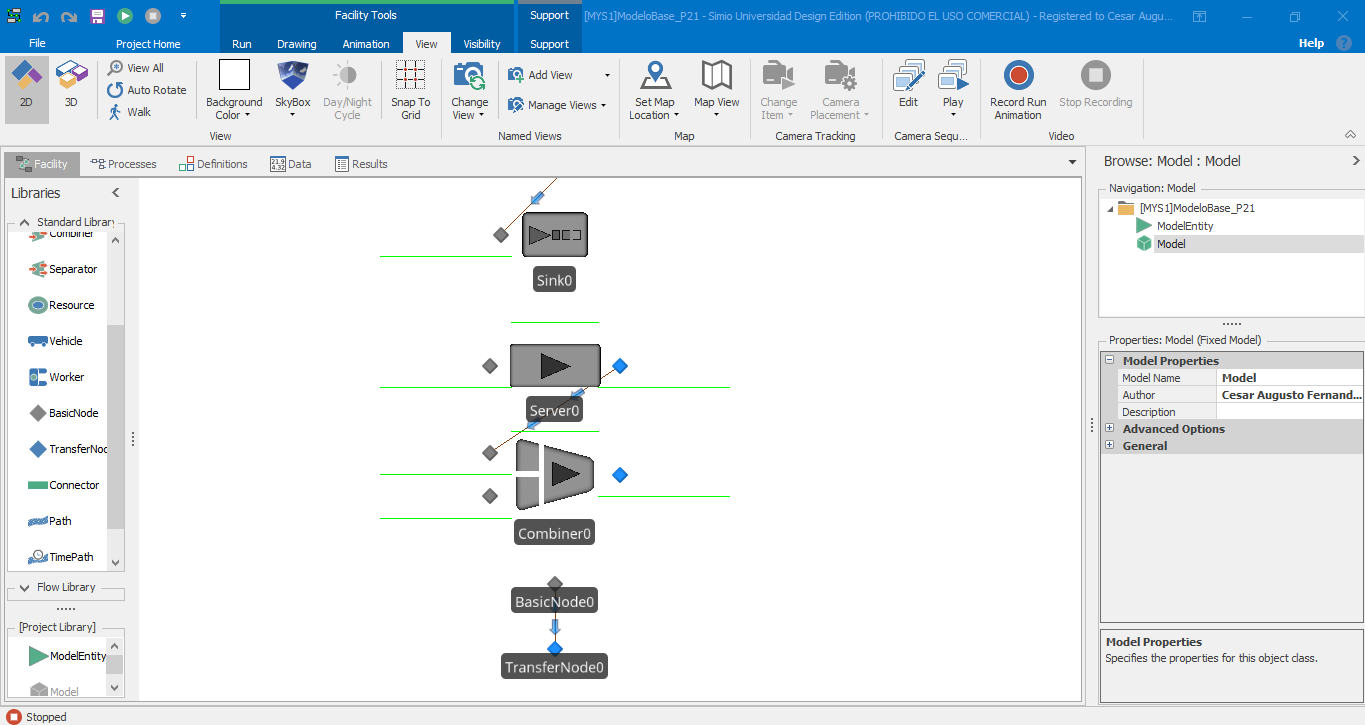
Se abre visual studio 2019 y se crea un nuevo programa y se elige la opción Aplicación de Windows Form.





## Paso 3

Se crea un modelo base que se necesitara para crear sobre el el modelo final. En este modelo se colocará todos los componentes que se utilizaran para llegar al modelo final.



## Paso 4

Se incorporan a visual studio 2019 los componentes que necesita para poder genera una api que pueda crear un documento con extensión de simio el cual tendrá nuestro modelo final. Estas librerías están en la carpeta donde esta instalado simio. Luego las colocamos el la carpeta debug de nuestro proyecto de visual

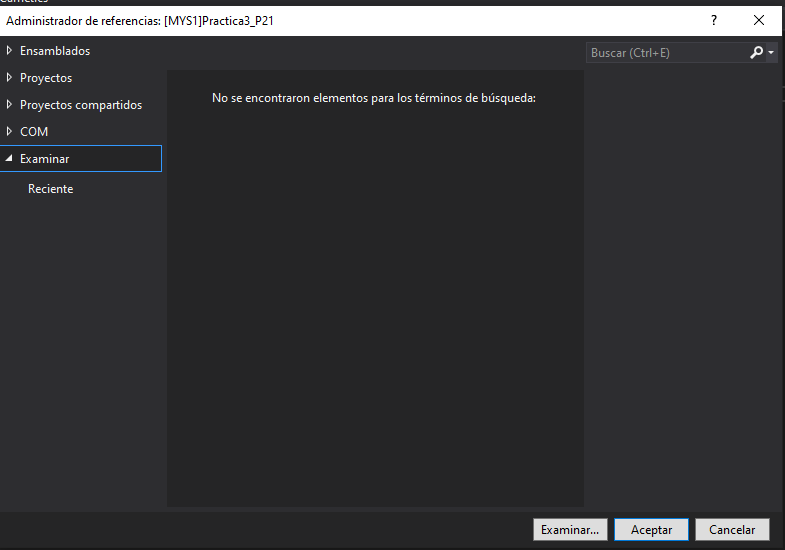


## Paso 5

En visual studio se agregan los componentes y librerías que se acaban de colocar en la carpeta debug de nuestro proyecto.

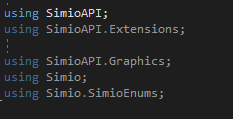
### Paso 5.1

Se agrega una nueva referencia y seleccionamos los componentes y librerías que están en nuestra carpeta debug



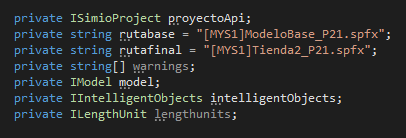
## Paso 6

Se agregan las librerías a nuestra clase de C#



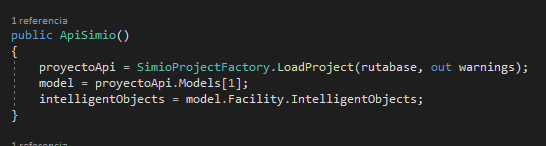
## Paso 7

Se manda a llamar las librerías a utilizar y las que nos ayudaran a genera nuestro modelo final.



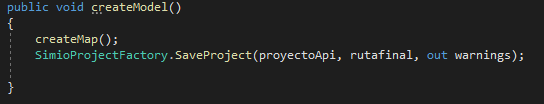
## Paso 8

Acá es donde creamos un método para poder llamarlo en cualquier parte de nuestro código



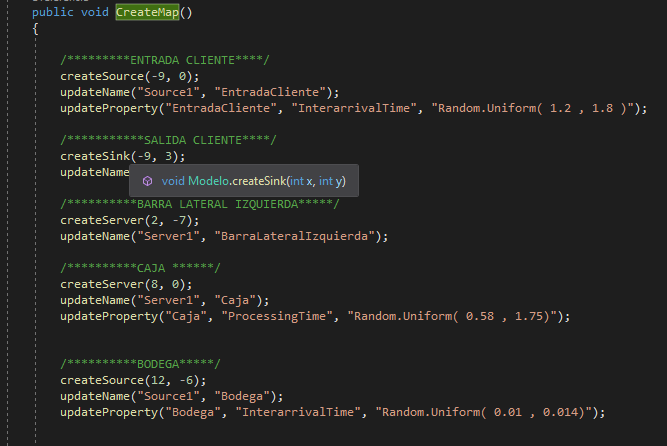
# Descripción de las funciones mas importantes de visual studio

## CreateModel

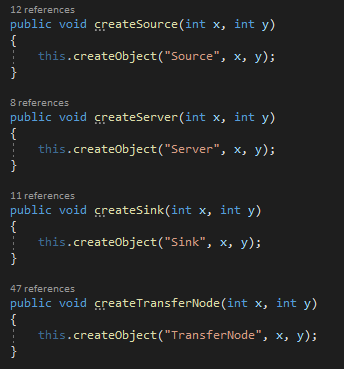
Este método nos sirve para mandar a llamar el método de createMap, además nos sirve para guardar la versión final del proyecto.

## createMap

Este método nos sirve para mandar a llamar el método de creación de todo el modelo como tal.

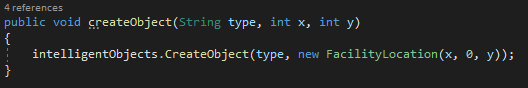


## createSource, createServer, createSink y createTransferNode

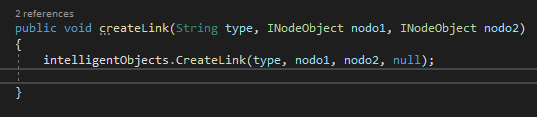
Estos métodos nos sirven para llamar el método para crear objetos, enviándole como parámetros el tipo de objeto, su posición en X y su posicion en Y.

## createObject

Este método recibe la llamada de los métodos descritos anteriormente con sus parámetros y utiliza la función CreateObject de intelligentObjects para crear el componente solicitado.

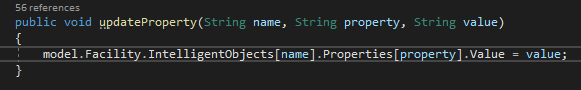


## createLink

Este método recibe la llamada de los métodos descritos anteriormente con sus parámetros y utiliza la función CreateLink de intelligentObjects para crear el enlace solicitado.

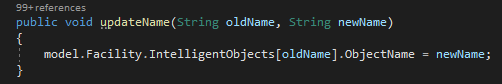
## updateProperty

Este método nos sirve para modificar las propiedades de los diferentes componentes utilizados en el modelo. Recibe como parámetros el nombre del componente o enlace, la propiedad a modificar y el nuevo valor a setear.

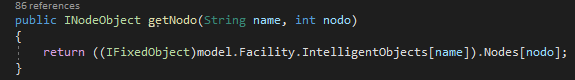


## updateName

Este método es parecido al anterior solo que este únicamente sirve para modificar el nombre del componente o enlace a cambiar. Recibe los parámetros de nombre antiguo y nombre nuevo.

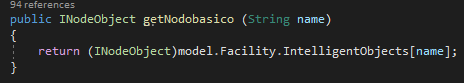


## getNodo

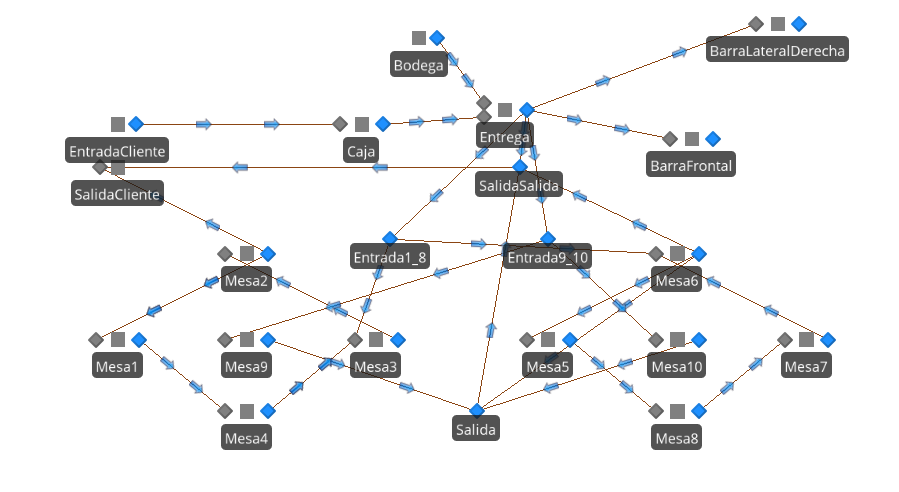
Este método sirve para obtener el nodo de un componente que servirá de enlace hacia otro componente o transfernode.

## getNodobasico

Este método sirve para obtener el nodo de un transfernode que servirá de enlace con otro componente o transfernode.



## Modelo Final



En este modelo final para la Tienda 2 se hizo esta distribución para que tenga una mejor visión de las entidades que van recorriendo dentro del modelo.

## Mejoras al sistema

* Elaboración de Vinos
  + Tener un poco mas de tanques de vinos con menos capacidad y así poder sacar un tanque que ya haya cumplido con el tiempo de fragmentación y no parte por parte
* Tienda 5
  + Colocar mas sillas en la barra y no tanta mesa
  + Tener un mas trabajadores en el área de bodega para que se reduzca el tiempo de sacar las botellas.
* API (Tienda 2)
  + Reubicar las mesas para que se pueda aprovechar todo el espacio posible
  + Tener mesas más grandes y que tengan mayor capacidad.

## Conclusiones

* Para la elaboración de vino es un proceso muy tardo en el cual si se necesita tener un plan para dicha preparación
* Tienda 5 Y API (Tienda 2) : Reubicar las áreas de espera ya que se pierde tiempo al momento de movilizarse dentro de la tienda.