

# Negocio

Se trata de una fábrica de bebidas. Actualmente solo se dedica a fabricar cervezas de dos tipos: IPA y Kolsh.

El usuario debe ingresar en el formulario la cantidad de litros de cerveza que desea preparar y su tipo. De acuerdo a esto, el programa se encarga de calcular los ingredientes y la cantidad necesaria de los mismos.

Una vez obtenida la receta, para iniciar el proceso de preparación, el programa valida que:

- Hay stock suficiente de cada uno de los ingredientes necesarios
- Hay espacio disponible en cada uno de los fermentadores teniendo en cuenta su capacidad y tipo de cerveza que almacena.

Si se dan estas condiciones, se empieza a cocinar la cerveza, se descuentan los ingredientes del stock actual, y se suma al inventario de cervezas las que están en proceso de preparación.

Queda fuera de scope otras de las instancias de fabricación de este producto, como el enfriado, envasado, y embotellado.

## Donde aplique los temas?

### Unit Test

Se utilizan unit test en el proyecto CervezaArtesanalTest. En el mismo se implementan test unitarios para probar los métodos de las clases FabricaBebidas y Receta.

### Generics

Se implementan generics en las clases Texto y XML (ver [aca](#)). A su vez la interfaz que implementan es genérica.

### Interfaces

Tanto la clase Texto como la clase XML implementan la interfaz IArchivo, con el método Guardar.

La clase Receta implemente IReceta.

### Excepciones

Se implementa una clase [RecetaException](#). Esta, al igual que otras excepciones son catcheadas en el método Cocinar de la clase [FabricaBebidas](#).

## Archivos

Se implementan las clases genéricas XML y Texto con el método Guardar. Los errores que se presenten van a ser logueados en un archivo de txt, mientras que la cerveza que se pueda empezar a cocinar se va a guardar en el xml.

## Diagrama de clases

