

Proyecto 3

Fábrica de cerveza



Mi nombre es Abel Alejandro Pacheco Quispe soy estudiante de ingeniería en Sistemas con experiencia en desarrollo de software, especializado en Java y Python. He participado en proyectos como una aplicación de inventarios, una red social de fotos y un software de rutas óptimas, demostrando habilidades en control de existencias, manejo de rutas de viaje y registro avanzado de usuarios. Con una formación sólida y conocimientos en redes, bases de datos SQL y NoSQL, y herramientas como GitHub, me destaco por mi capacidad de adaptación, resolución de problemas y compromiso con la calidad. Busco un entorno dinámico que me permita seguir desarrollando mis competencias y aportar con soluciones innovadoras.

Nombre: Abel Alejandro Pacheco Quispe

Materia: Taller de Simulación de Sistemas

Docente: Ayoroa Cardozo Jose Richard

Contenido

| | |
|---|----|
| Definición del sistema: | 3 |
| Colección de datos: | 4 |
| Formulación del modelo: | 5 |
| • LEAP | 5 |
| • Diagrama de simulación: | 7 |
| • Diagrama de simulación ProModel: | 8 |
| Implementación del modelo en computadora: | 8 |
| ProModel: | 8 |
| Java: | 9 |
| Validación: | 9 |
| Forma manual: | 9 |
| Pro model: | 10 |
| Java: | 11 |
| Tabla de comparación: | 11 |
| Experimentación: | 12 |
| Pro Model: | 12 |
| Java: | 14 |
| Interpretación: | 16 |
| Pro Model: | 16 |
| Graficas Pro Model: | 17 |
| Java: | 18 |
| Graficas Java: | 20 |
| Documentación: | 20 |
| Archivo de texto Pro Model: | 20 |
| Código de java: | 24 |
| Resultados: | 35 |
| Pro Model: | 35 |
| Java: | 37 |
| Tabla de comparación: | 39 |
| Conclusión: | 40 |
| Videos del proyecto: | 40 |

Definición del sistema:

En una empresa productora de cerveza artesanal arriba 1 kg de granos de cebada cada 25 minutos, los cuales reciben en un silo con capacidad para 3 kg los granos, después pasan a otro silo en donde se realiza el proceso de malteado o germinado de los granos, este proceso dura 1 hora y el silo de malteado tiene capacidad para 3 kg de granos, posteriormente estos granos pasan a una estación de secado, la cual tiene capacidad para 3 kg y el proceso de secado tiene una duración de 1 hora. Los granos pasan después a una locación de molienda, la cual tiene capacidad para 2 kg de granos y cuyo proceso dura 1 hora. Un operador se encarga de trasladar las unidades de la locación de malteado a la de secado y luego de la locación de secado hasta la de molienda, a 90 pasos por minuto y luego de que entrega se devuelve al silo de lupulo.

Los granos pasan a una locación de macerado, cuyo proceso tiene una duración de 1,5 horas y tiene capacidad para procesar 3 kg. Luego los granos pasan a una locación de filtrado en donde se separan los granos mezclados con agua de las partículas existentes, este proceso tiene una duración de 30 minutos y la locación tiene capacidad para 2 kg. Finalmente, los granos de cebada son trasladados a una locación de cocción, en donde se hierven los granos durante 1 hora. Dicha locación tiene capacidad para 10 litros.

Por otro lado, a otra locación arriban 1 kg de lupulo cada 10 minutos, los cuales son almacenados ya que la locación tiene capacidad para 10 kg. El lupulo es trasladado por un operador a la locación de cocción. Este operador se mueve a 100 pasos por minuto. Posteriormente en la locación de cocción, se unen los granos molidos de cebada con 4 kg de lupulo para formar una mezcla conocida como Mosto.

Luego de la cocción se lleva el mosto a una locación para su enfriamiento, el cual es un proceso que tiene una duración de 1 hora y la maquina tiene capacidad para enfriar y almacenar 10 litros de líquido. Seguidamente la mezcla pasa a una locación de fermentación en donde espera a ser unida con levadura. En este sentido, en otra locación con capacidad de 10 kg, arriba 1 kg de levadura cada 20 minutos, la cual es trasladada por un operador a la locación de fermentación cuya capacidad es 10 litros, tomando en cuenta que el operador se mueve a 100 pasos por minuto y luego de que entra se devuelve al silo de levadura. En la locación de fermentación se unen 10 litros de mosto con 2 kilogramos de levadura para formar el alcohol de cerveza y CO₂, este proceso tiene una duración de 2 horas.

El líquido de cerveza es trasladado a una locación de maduración, en donde se deja reposar por 1,5 horas y la locación tiene capacidad para almacenar 10 litros de cerveza. Posteriormente la cerveza pasa a una locación de inspección la cual tiene la capacidad de analizar 3 litros de cerveza a la vez y existe el 90% de probabilidad de aprobar la muestra y el 10% de probabilidad de descartar la cerveza, la cual sale del sistema. Este proceso tiene una duración de 30 minutos. En el caso positivo y la muestra de cerveza es aprobada pasa a una locación de embotellado, la cual tiene capacidad y envasar 6 botellas a la vez y el proceso tiene una duración de 3 minutos. Seguidamente, las botellas de cerveza son trasladadas a una locación de etiquetado, la cual tiene capacidad de etiquetar 6 botellas a la vez y este proceso dura 1 minuto.

En otra locación, arriba una caja de cartón cada 30 minutos, la cual tiene capacidad para 30 cajas, las cuales son almacenadas para ser trasladadas a una locación de empacado la cual tiene la capacidad de empacar 6 botellas en una caja de cartón, quedando permanente unidas y denominado se cervezas empacadas. Esta operación tiene una duración de 10 minutos.

Las cervezas empacadas son trasladadas por un operador a una locación de almacenaje, la cual tiene capacidad de almacenar 6 cajas con cervezas. Este operador se mueve a 100 pasos por minuto y se devuelve al completar su tarea. El proceso de almacenaje tiene una duración de 5 minutos.

Finalmente, cuando el operador acumula 6 cajas, traslada las cajas de cerveza al mercado, utilizando un vehículo. Este mercado tiene capacidad infinita y representa la salida del sistema.

El sistema cuenta con 4 operadores y un vehículo, las locaciones solo tiene una unidad cada una. En el mismo se realizan 3 uniones, la de los granos filtrados con el lupulo para formar el mosto, la unión del mosto y la levadura para formar la cerveza y la unión de las 6 botellas y una caja de cartón para formar el producto terminado.

La empresa trabaja con una jornada laboral de 10 horas los 7 días a la semana. Se pide:

- Realizar la simulación del proceso de producción de cerveza durante una semana
- Analizar cuantas entidades fueron procesadas o salieron de cada locación,
- Cuantas se encuentran en proceso,
- Cuantas salieron del sistema
- Cuál es el tiempo promedio en el sistema.

Colección de datos:

Los datos que nos da el problema los podemos ver en la siguiente tabla:

| Etapa/Elemento | Detalle |
|-----------------------------------|---|
| Recepción de Materias Primas | |
| Cebada | Arribo: 1 kg cada 25 min. Almacenamiento: silo de 3 kg. |
| Lúpulo | Arribo: 1 kg cada 10 min. Almacenamiento: silo de 10 kg. |
| Levadura | Arribo: 1 kg cada 20 min. Almacenamiento: silo de 10 kg. |
| Producción - Procesos Iniciales | |
| Malteado | Capacidad: 3 kg. Duración: 1 hora. |
| Secado | Capacidad: 3 kg. Duración: 1 hora. |
| Molienda | Capacidad: 2 kg. Duración: 1 hora. |
| Macerado | Capacidad: 3 kg. Duración: 1,5 horas. |
| Filtrado | Capacidad: 2 kg. Duración: 30 minutos. |
| Cocción y Mezcla | |
| Cocción | Capacidad: 10 litros. Duración: 1 hora. Mezcla: granos molidos + 4 kg de lúpulo. |
| Enfriamiento | Capacidad: 10 litros. Duración: 1 hora. |
| Fermentación | Capacidad: 10 litros. Duración: 2 horas. Mezcla: 10 litros de mosto + 2 kg de levadura. |
| Maduración | Capacidad: 10 litros. Duración: 1,5 horas. |
| Inspección de Calidad | Capacidad: 3 litros. Duración: 30 minutos. Probabilidad de aprobación: 90% |
| Embotellado, Etiquetado y Empaque | |
| Embotellado | Capacidad: 6 botellas por ciclo. Duración: 3 minutos. |

| | |
|-----------------------------------|---|
| Etiquetado | Capacidad: 6 botellas por ciclo. Duración: 1 minuto. |
| Empaque | Capacidad: 6 botellas por caja. Duración: 10 minutos. |
| Cajas | Arribo: 1 caja cada 30 min. Almacenamiento: 30 cajas. |
| Almacenamiento y Transporte Final | |
| Almacenamiento de Cajas | Capacidad: 6 cajas. Duración: 5 minutos. |
| Transporte al Mercado | Vehículo: 6 cajas por viaje. Capacidad del mercado: infinita. |
| Operadores y Movimientos | |
| Número de Operadores | 4 |
| Velocidad Operadores | Cebada/Lúpulo: 90 pasos/min. Levadura: 100 pasos/min. |
| Uniones del Sistema | |
| Mosto | Granos filtrados + 4 kg de lúpulo. |
| Cerveza | 10 litros de mosto + 2 kg de levadura. |
| Producto Final | 6 botellas etiquetadas + 1 caja de cartón (producto terminado). |

Formulación del modelo:

- LEAP

Locaciones:

- **Silo cebado:** capacidad = 3; unidades = 1.
- **Malteado:** capacidad = 3; unidades = 1.
- **Secado:** capacidad = 3; unidades = 1.
- **Molienda:** capacidad = 2; unidades = 1.
- **Macerado:** capacidad = 3; unidades = 1.
- **Filtrado:** capacidad = 2; unidades = 1.
- **Cocción:** capacidad = 10; unidades = 1.
- **Silo lupulo:** capacidad = 10; unidades = 1.
- **Enfriado:** capacidad = 10; unidades = 1.
- **Silo levadura:** capacidad = 10; unidades = 1.
- **Fermentado:** capacidad = 10; unidades = 1.
- **Maduración:** capacidad = 10; unidades = 1.
- **Inspección:** capacidad = 3; unidades = 1.
- **Embotellado:** capacidad = 6; unidades = 1.
- **Etiquetado:** capacidad = 6; unidades = 1.
- **Almacén de cajas:** capacidad = 30; unidades = 1.
- **Empacado:** capacidad = 6; unidades = 1.
- **Almacenaje:** capacidad = 6; unidades = 1.
- **Mercado:** capacidad = Infinito; unidades = 1.

Entidades:

- Grano de cebada.
- Lupulo.
- Mosto.
- Levadura.
- Alcohol.
- Botella de cerveza.
- Caja bacía.
- Caja con cerveza.

Arribos/Llegadas:**Arribos a silo de cebada:**

- Entidad = Grano de cebada
- Locación = Silo cebada
- Capacidad Arribo = 1
- Primera vez = 0
- Ocurrencia = Infinito
- Frecuencia = 25 min

Arribos a silo de levadura:

- Entidad = Levadura
- Locación = Silo levadura
- Capacidad Arribo = 1
- Primera vez = 0
- Ocurrencia = Infinito
- Frecuencia = 20 min

Arribos a silo de lupulo:

- Entidad = Lupulo
- Locación = Silo lupulo
- Capacidad Arribo = 1
- Primera vez = 0
- Ocurrencia = Infinito
- Frecuencia = 10 min

Arribos a almacén de cajas:

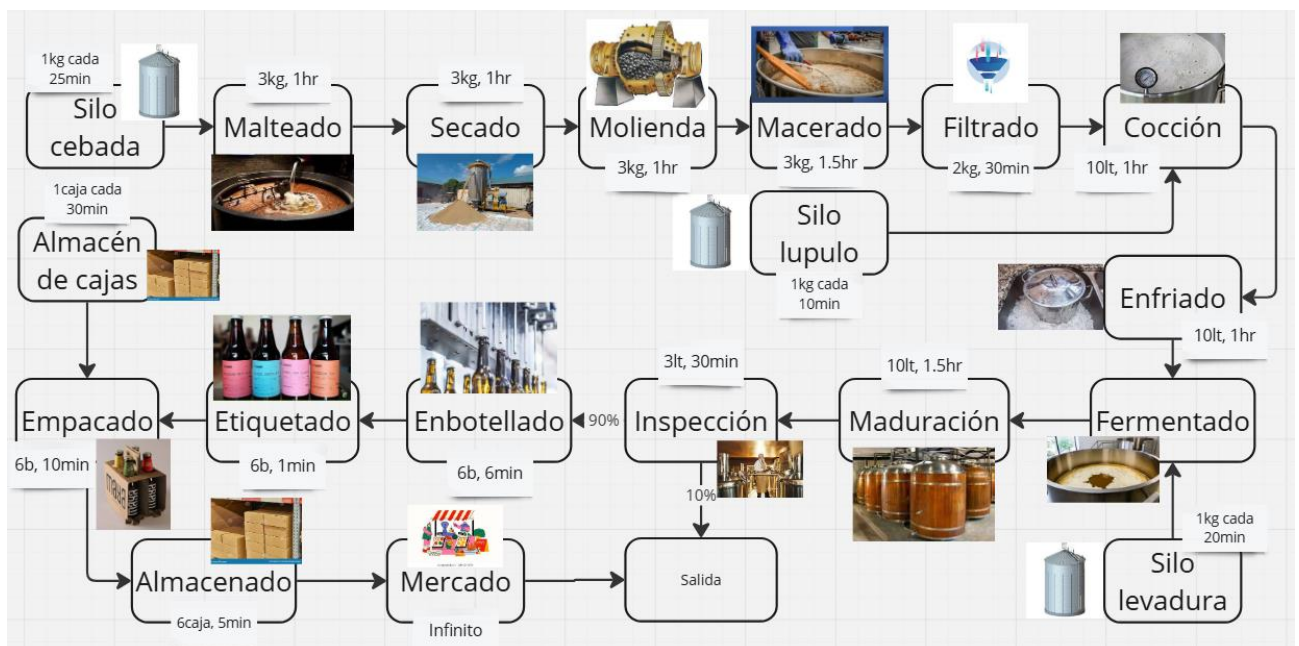
- Entidad = Caja bacía
- Locación = almacén de cajas
- Capacidad Arribo = 1
- Primera vez = 0
- Ocurrencia = Infinito
- Frecuencia = 30 min

Procesos:

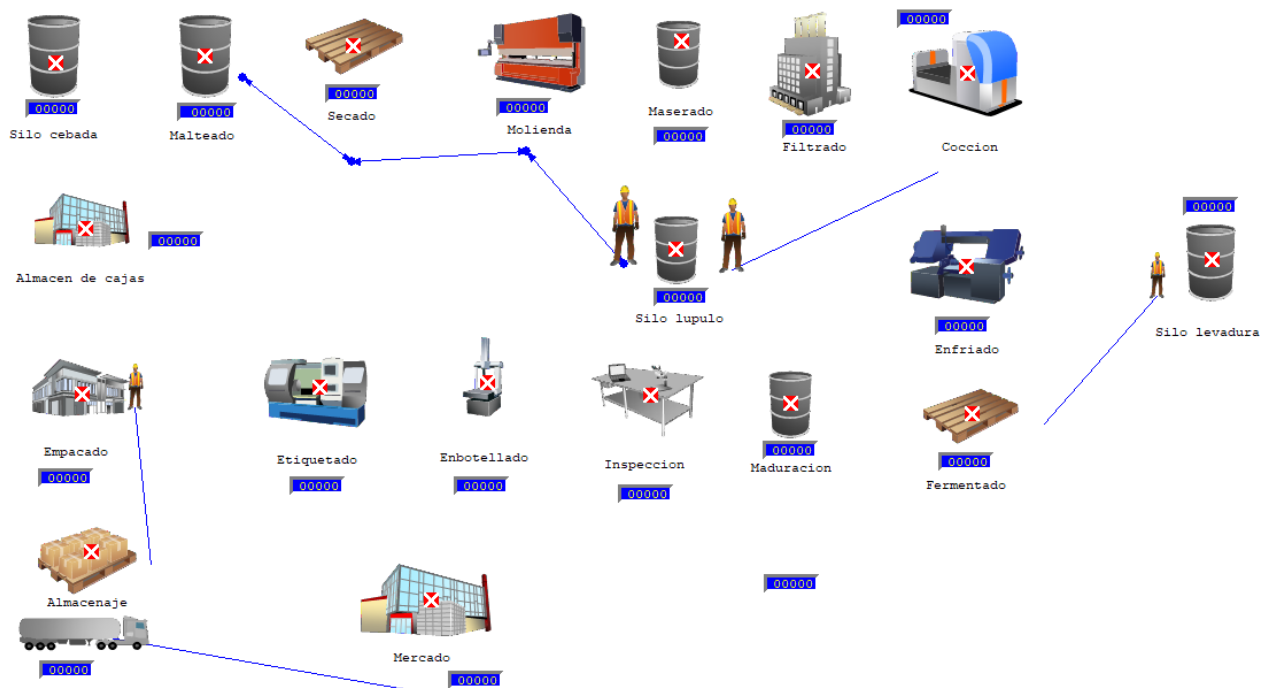
1. Grano esta en silo de cebada y pasa a área de malteado.
2. Grano está en el área de malteado y espera 1 hora luego pasa a área de secado utilizando el operador de recepción.
3. Grano está en área de secado y espera 1 hora luego pasa a área de molienda utilizando el operador de recepción.
4. Grano está en área de molienda y espera 1 hora luego pasa a área de macerado.
5. Grano está en área de macerado y espera 1.5 horas luego pasa a área de filtrado.
6. Grano está en área de filtrado y espera 0.5 horas luego pasa a área de cocción.
7. **Lupulo esta en silo de lupulo y espera a que se la necesite para que el operador de lupulo la lleve.**

8. Grano está en área de cocción y espera que se mezcle con 2 lupulos para esperar 1 hora y pasar a enfriado como mosto.
9. Mosto está en el área de enfriado y espera 1 hora y luego para al área de fermentado.
- 10. Levadura está en el silo de levadura y espera a que se la necesite para que el operador de levadura la lleve.**
11. Mosto está en el área de fermentado se combinan 10 mosto y 2 levaduras para esperar 1 hora y luego pasar al área de maduración como alcohol.
12. Alcohol está en área de maduración un tiempo de 1.5 horas y luego pasa al área de inspección.
13. Alcohol está en inspección espera un tiempo de 0.5 horas y tiene un 90% de probabilidad de ser aprobado y un 10% de ser descartado, en caso de ser aprobado pasa a área de Embotellado.
14. Alcohol está en área de embotellado espera un tiempo de 6 minutos y pasa al área de etiquetado como botella con cerveza.
15. Botella de cerveza está en área de etiquetado y espera un tiempo de 1 minuto y luego pasa al área de empacado.
- 16. Caja bacia esta en el almacén de cajas y espera a que se la necesite para que un operador la lleve.**
17. Botella de cerveza está en el área de empacado de combina con 6 y se une a una caja y espera un tiempo de 10 minutos luego pasa a almacenaje como caja con cerveza.
18. Caja con cerveza esta en almacenaje y espera un tiempo de 5 minutos luego pasa al área de mercado utilizando el camión.
19. Caja con cerveza esta en el mercado y sale.

• **Diagrama de simulación:**



- Diagrama de simulación ProModel:



Implementación del modelo en computadora:

ProModel y Java son herramientas útiles en el desarrollo de simulaciones y la creación de interfaces gráficas para visualizar resultados, cada una con características que se adaptan a diferentes necesidades en proyectos de simulación y modelado.

ProModel:

ProModel es un software especializado en la simulación de procesos y sistemas. Se utiliza principalmente para modelar, analizar y optimizar procesos complejos de flujo, como los que se encuentran en la manufactura, logística, salud y servicios. Su diseño visual permite modelar de manera rápida y precisa escenarios de procesos en los que el flujo de recursos y la toma de decisiones son clave para identificar cuellos de botella y mejorar la eficiencia.

Características Principales:

- **Modelado visual:** Permite crear modelos visuales e interactivos que representan el flujo de trabajo.
- **Optimización:** Cuenta con herramientas para evaluar diferentes escenarios y encontrar la configuración óptima para el proceso.
- **Análisis de resultados:** Genera reportes detallados y gráficos que ayudan en la interpretación de los resultados de la simulación.

- **Interfaz amigable:** Ofrece un entorno intuitivo, especialmente para usuarios con poca experiencia en programación.

Java:

Java es un lenguaje de programación versátil que permite desarrollar aplicaciones completas con interfaces gráficas personalizadas y lógica de negocio robusta. En el contexto de simulaciones, Java es útil para crear aplicaciones que requieran una interfaz de usuario y operaciones matemáticas complejas, así como para personalizar funcionalidades de la simulación que no son posibles o fáciles de implementar en herramientas especializadas como ProModel.

Características Principales:

- **Interfaz gráfica (GUI):** Java, junto con librerías como Swing y JavaFX, permite desarrollar aplicaciones con interfaces gráficas que facilitan la interacción con los datos de simulación.
- **Portabilidad:** Al ser un lenguaje multiplataforma, las aplicaciones desarrolladas en Java pueden ejecutarse en diferentes sistemas operativos sin modificaciones.
- **Integración de cálculos personalizados:** Java permite implementar cálculos específicos y algoritmos personalizados que no siempre se pueden realizar en software de simulación estándar.
- **Extensibilidad:** Ofrece la flexibilidad de integrar módulos adicionales y mejorar la funcionalidad según las necesidades del proyecto.

Validación:

Forma manual:

Para la validación vamos a realizar un proceso de simulación manual para ver las cantidades de entidades que existen en las primeras locaciones en un periodo de 5 horas esto para poder ver su comportamiento y debido a que las cantidades que maneja el modelo en sus primeras locaciones determinan cuantas saldrán al final.

| Silo de cebada | Malteado | Secado | Molienda | Maserado | Filtrado |
|----------------|----------|--------|----------|----------|----------|
| | 00:25 | | | | |
| | 00:50 | 01:35 | | | |
| | 01:15 | 01:50 | 02:25 | | |
| | 01:40 | 02:15 | 02:50 | 03:25 | |
| | 02:10 | 02:40 | 03:15 | 03:50 | 04:25 |
| | 02:35 | 03:10 | 03:40 | 04:15 | 04:50 |
| | 03:00 | 03:35 | 04:10 | 04:40 | |
| | 03:25 | 04:00 | 04:35 | | |
| | 03:50 | 04:25 | | | |
| | 04:15 | 04:50 | | | |
| | 04:40 | | | | |
| Total: | 11 | 9 | 6 | 4 | 2 |

Como se puede ver en la imagen tenemos un proceso de las primeras locaciones donde entra la cebada en el 0:25 y sigue su proceso hasta el área de filtrado en hora 4:25.

Esto se puede hacer debido a que las locaciones tardan 1 hora en procesar cada kilogramo de cebada por lo que entran y salen, también vemos que cumple con las capacidades donde si un cuarto quiere entrar tiene que esperar a que el primero que entro salga eso se hace viendo que la hora siguiente sea menor a la hora del que va a entrar.

Y si hacemos un conteo de las entidades que pasaron en total tenemos que:

Por malteado pasaron 11, por secado pasaron 9, por molienda pasaron 6, por macerado 4 y por filtrado 2, eso en un periodo de 5 horas.

Y de igual forma para los arribos tenemos que en el silo de cebada, en el silo de lúpulo, en el silo de levadura y en el almacén de cajas haciendo cálculos tenemos que.

En silo cebada si 300 minutos (5 horas) lo dividimos entre el intervalo de llegada de 25 minutos tenemos que para ese momento llegaron 12 kilogramos de cebada.

En el silo de lupulo si 300 minutos (5 horas) lo dividimos entre su intervalo de llegada que es de 10 min tenemos que para ese momento llegaron 30 kilogramos de lupulo pero por la capacidad solo se perdieron 20 kilogramos de los 30.

En el silo de levadura si 300 minutos (5 horas) lo dividimos entre su intervalo de llegad de 20 minutos tenemos que llegaron 15 kilogramos, pero por la capacidad se desecharon 5.

En el almacén de cajas si en 300 minutos (5 horas) lo dividimos entre su intervalo de 30 minutos tenemos que en ese tiempo llegaron 10 cajas.

Pro model:

Para la simulación en pro model tenemos que en 5 horas.

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 13,00 | 23,08 | 0,00 | 13,00 | 13,00 | 6,50 |
| contMalteado | 12,00 | 22,92 | 0,00 | 12,00 | 12,00 | 6,50 |
| contSecado | 9,00 | 30,33 | 0,00 | 9,00 | 9,00 | 4,11 |
| contMolieda | 6,00 | 45,38 | 0,00 | 6,00 | 6,00 | 2,09 |
| contMaserado | 4,00 | 67,90 | 0,00 | 4,00 | 4,00 | 0,99 |
| contFiltrado | 2,00 | 148,29 | 0,00 | 2,00 | 2,00 | 0,11 |
| contCoccion | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contSiloLupulo | 10,00 | 9,00 | 0,00 | 10,00 | 10,00 | 8,50 |
| contEnfriado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contSiloLevadura | 10,00 | 18,00 | 0,00 | 10,00 | 10,00 | 7,00 |
| contFermentado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contMaduracion | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contInspeccion | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contRechasada | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contEnbotellado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contEtiquetado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contEnpacado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contAlmacenCajas | 11,00 | 27,27 | 0,00 | 11,00 | 11,00 | 5,50 |

La cantidad que pasaron por el silo es de 13 kilogramos, por el área de malteado es de 12 kilogramos, por el área de secado es de 9 kilogramos, por el área de molienda es de 6 kilogramos, por el área de macerado es de 4 kilogramos y el área de filtrado es de 2 kilogramos.

Mientras que para los silos de cebada tenemos que el total de entradas es de 13 kilogramos, para el silo de lupulo es de 10 kilogramos, para el silo de levadura es de 10 kilogramos y para el almacén de cajas es de 11 cajas en un periodo de 5 horas.

Java:

Para la simulación en Java tenemos que en 5 horas.

| Descripción | Valor |
|---|-------|
| Arribos | |
| Arribo de cebada | 12 |
| Arribo fallido de cebada | 0 |
| Arribo de lupulo | 10 |
| Arribo fallido de lupulo | 20 |
| Arribo de levadura | 10 |
| Arribo fallido de levadura | 6 |
| Arribo de caja vacia | 10 |
| Arribo fallido de caja vacia | 0 |
| Cantidades procesadas por cada locacion | |
| En malteado | 12 |
| En secado | 9 |
| En molienda | 6 |
| En maserado | 4 |
| En filtrado | 2 |
| En cocción | 0 |
| En enfriado | 0 |

Como se puede ver para la simulación de in periodo de 5 horas en java tenemos que la cantidad kilogramos de cebada que paso por el silo es de 12, la cantidad de kg de lúpulo que paso por el silo de lúpulo es de 10, la cantidad de levadura que paso por el silo de levadura es de 10 kg, y la cantidad de cajas que entraron al almacén de cajas es de 10 cajas, mientras que para las locaciones tenemos que para malteado entraron 12 kg, para secado entraron 9 kg, para macerado entraron 4 kg y para filtrado entraron 2 kg.

Tabla de comparación:

| Total, entradas | Forma manual | Pro model | Java |
|------------------|--------------|-----------|----------|
| Malteado | 11 kg | 12 kg | 12 kg |
| Secado | 9 kg | 9 kg | 9 kg |
| Molienda | 6 kg | 6 kg | 6 kg |
| Macerado | 4 kg | 4 kg | 4 kg |
| Filtrado | 2 kg | 2 kg | 2 kg |
| Silo de cebada | 12 kg | 13 kg | 12 Kg |
| Silo de lupulo | 10 kg | 10 kg | 10 kg |
| Silo de levadura | 10 kg | 10 kg | 10 kg |
| Almacén de cajas | 10 cajas | 11 cajas | 10 cajas |

Como se puede ver el modelo cumple con los tiempos establecidos para los arribos y los tiempos de proceso para las locaciones principales sin haber diferencias significativas entre los diferentes

tipos de modelo por lo que podemos decir que el modelo es válido y es una representación de la realidad del problema señalado.

Experimentación:

Para la experimentación probaremos modificando el punto donde las entradas totales disminuyen de forma exponencial.

Y con las cantidades de las primeras áreas de proceso que son las áreas de silo de grano, área de malteado, secado, molienda, macerado y filtrado.

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 124,00 | 33,87 | 0,00 | 124,00 | 124,00 | 68,02 |
| contMalteado | 121,00 | 34,56 | 0,00 | 121,00 | 121,00 | 65,68 |
| contSecado | 118,00 | 35,44 | 0,00 | 118,00 | 118,00 | 62,72 |
| contMolieda | 115,00 | 36,36 | 0,00 | 115,00 | 115,00 | 59,81 |
| contMaserado | 113,00 | 36,99 | 0,00 | 113,00 | 113,00 | 57,89 |
| contFiltrado | 110,00 | 38,00 | 0,00 | 110,00 | 110,00 | 55,05 |
| contCoccion | 108,00 | 38,71 | 0,00 | 108,00 | 108,00 | 53,51 |
| contSiloLupulo | 400,00 | 10,50 | 0,00 | 400,00 | 400,00 | 190,50 |
| contEnfriado | 98,00 | 42,66 | 0,00 | 98,00 | 98,00 | 45,67 |
| contSiloLevadura | 28,00 | 141,43 | 0,00 | 28,00 | 28,00 | 17,71 |
| contFermentado | 96,00 | 43,34 | 0,00 | 96,00 | 96,00 | 44,02 |
| contMaduracion | 9,00 | 448,99 | 0,00 | 9,00 | 9,00 | 3,77 |
| contInspeccion | 9,00 | 458,99 | 0,00 | 9,00 | 9,00 | 3,58 |
| contRechasada | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contEnbotellado | 9,00 | 462,32 | 0,00 | 9,00 | 9,00 | 3,51 |
| contEtiquetado | 9,00 | 462,99 | 0,00 | 9,00 | 9,00 | 3,50 |
| contEnpackado | 9,00 | 463,10 | 0,00 | 9,00 | 9,00 | 3,50 |
| contAlmacenCajas | 31,00 | 95,81 | 0,00 | 31,00 | 31,00 | 27,19 |
| contAlmacen | 1,00 | 2.978,12 | 0,00 | 1,00 | 1,00 | 0,29 |
| contMercado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |

Pro Model:

- Prueba 1: En el proceso de fermentación cambiamos la cantidad de 10 litros de mosto y 2 kilogramos de levadura a 5 litros y 2 de levadura.

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 124,00 | 33,87 | 0,00 | 124,00 | 124,00 | 68,02 |
| contMalteado | 121,00 | 34,56 | 0,00 | 121,00 | 121,00 | 65,68 |
| contSecado | 118,00 | 35,44 | 0,00 | 118,00 | 118,00 | 62,72 |
| contMolieda | 115,00 | 36,36 | 0,00 | 115,00 | 115,00 | 59,81 |
| contMaserado | 113,00 | 36,99 | 0,00 | 113,00 | 113,00 | 57,89 |
| contFiltrado | 110,00 | 38,00 | 0,00 | 110,00 | 110,00 | 55,05 |
| contCoccion | 108,00 | 38,71 | 0,00 | 108,00 | 108,00 | 53,51 |
| contSiloLupulo | 400,00 | 10,50 | 0,00 | 400,00 | 400,00 | 190,50 |
| contEnfriado | 98,00 | 42,66 | 0,00 | 98,00 | 98,00 | 45,67 |
| contSiloLevadura | 48,00 | 86,67 | 0,00 | 48,00 | 48,00 | 26,51 |
| contFermentado | 96,00 | 43,34 | 0,00 | 96,00 | 96,00 | 44,28 |
| contMaduracion | 18,00 | 224,50 | 0,00 | 18,00 | 18,00 | 7,96 |
| contInspeccion | 18,00 | 229,50 | 0,00 | 18,00 | 18,00 | 7,58 |
| contRechasada | 1,00 | 3.360,92 | 0,00 | 1,00 | 1,00 | 0,20 |
| contEnbotellado | 17,00 | 244,76 | 0,00 | 17,00 | 17,00 | 7,25 |
| contEtiquetado | 17,00 | 245,11 | 0,00 | 17,00 | 17,00 | 7,22 |
| contEnpackado | 17,00 | 245,17 | 0,00 | 17,00 | 17,00 | 7,22 |
| contAlmacenCajas | 32,00 | 92,81 | 0,00 | 32,00 | 32,00 | 27,76 |
| contAlmacen | 2,00 | 1.489,06 | 0,00 | 2,00 | 2,00 | 0,87 |
| contMercado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |

Como se puede ver al reducir los litros utilizados en fermentación se duplican las salidas en maduración para producir más cervezas, pero de igual forma vemos que en el área de mercado no llevo nada debido a que se necesita 6 cajas con cerveza para que el camión pueda llevar las cervezas al mercado.

- Prueba 2: En el proceso de fermentación cambiamos la cantidad de 10 litros de mosto y 2 kilogramos de levadura a 1 litros y 2 de levadura.

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 124,00 | 33,87 | 0,00 | 124,00 | 124,00 | 68,02 |
| contMalteado | 121,00 | 34,56 | 0,00 | 121,00 | 121,00 | 65,68 |
| contSecado | 118,00 | 35,44 | 0,00 | 118,00 | 118,00 | 62,72 |
| contMolieda | 115,00 | 36,36 | 0,00 | 115,00 | 115,00 | 59,81 |
| contMaserado | 113,00 | 36,99 | 0,00 | 113,00 | 113,00 | 57,89 |
| contFiltrado | 110,00 | 38,00 | 0,00 | 110,00 | 110,00 | 55,05 |
| contCoccion | 108,00 | 38,71 | 0,00 | 108,00 | 108,00 | 53,51 |
| contSiloLupulo | 400,00 | 10,50 | 0,00 | 400,00 | 400,00 | 190,50 |
| contEnfriado | 98,00 | 42,66 | 0,00 | 98,00 | 98,00 | 45,67 |
| contSiloLevadura | 199,00 | 21,11 | 0,00 | 199,00 | 199,00 | 94,39 |
| contFermentado | 96,00 | 43,34 | 0,00 | 96,00 | 96,00 | 44,28 |
| contMaduracion | 93,00 | 44,74 | 0,00 | 93,00 | 93,00 | 41,55 |
| contInspeccion | 91,00 | 45,83 | 0,00 | 91,00 | 91,00 | 39,57 |
| contRechasada | 7,00 | 537,27 | 0,00 | 7,00 | 7,00 | 2,48 |
| contEnbotellado | 83,00 | 50,13 | 0,00 | 83,00 | 83,00 | 36,44 |
| contEtiquetado | 83,00 | 50,20 | 0,00 | 83,00 | 83,00 | 36,32 |
| contEnpackado | 83,00 | 50,22 | 0,00 | 83,00 | 83,00 | 36,30 |
| contAlmacenCajas | 43,00 | 92,79 | 0,00 | 43,00 | 43,00 | 32,53 |
| contAlmcen | 13,00 | 306,01 | 0,00 | 13,00 | 13,00 | 5,66 |
| contMercado | 2,00 | 1.851,70 | 0,00 | 2,00 | 2,00 | 0,63 |

Como se puede ver al reducir los litros utilizados en fermentación a 1 las salidas no tienen grandes cambios en maduración para producir más cervezas, pero de igual forma vemos que en el área de mercado esta vez si llegaron cajas con cerveza en este caso 2 cajas con cerveza lo cual no es mucho.

- Prueba 3: En el proceso de fermentación cambiamos la cantidad de 10 litros de mosto y 2 kilogramos de levadura a 1 litros y 2 de levadura y aumentando la capacidad de las primeras áreas más específicamente aumentando 1 las capacidades de cada una.

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 130,00 | 32,31 | 0,00 | 130,00 | 130,00 | 72,51 |
| contMalteado | 126,00 | 33,19 | 0,00 | 126,00 | 126,00 | 70,14 |
| contSecado | 122,00 | 34,28 | 0,00 | 122,00 | 122,00 | 66,64 |
| contMolieda | 118,00 | 35,43 | 0,00 | 118,00 | 118,00 | 63,15 |
| contMaserado | 115,00 | 36,35 | 0,00 | 115,00 | 115,00 | 60,40 |
| contFiltrado | 111,00 | 37,66 | 0,00 | 111,00 | 111,00 | 56,69 |
| contCoccion | 108,00 | 38,71 | 0,00 | 108,00 | 108,00 | 54,21 |
| contSiloLupulo | 400,00 | 10,50 | 0,00 | 400,00 | 400,00 | 190,50 |
| contEnfriado | 98,00 | 42,66 | 0,00 | 98,00 | 98,00 | 45,69 |
| contSiloLevadura | 199,00 | 21,11 | 0,00 | 199,00 | 199,00 | 94,39 |
| contFermentado | 96,00 | 43,34 | 0,00 | 96,00 | 96,00 | 44,30 |
| contMaduracion | 93,00 | 44,74 | 0,00 | 93,00 | 93,00 | 41,57 |
| contInspeccion | 91,00 | 45,83 | 0,00 | 91,00 | 91,00 | 39,59 |
| contRechasada | 7,00 | 537,27 | 0,00 | 7,00 | 7,00 | 2,48 |
| contEnbotellado | 83,00 | 50,13 | 0,00 | 83,00 | 83,00 | 36,46 |
| contEtiquetado | 83,00 | 50,20 | 0,00 | 83,00 | 83,00 | 36,34 |
| contEnpackado | 83,00 | 50,22 | 0,00 | 83,00 | 83,00 | 36,32 |
| contAlmacenCajas | 43,00 | 92,79 | 0,00 | 43,00 | 43,00 | 32,53 |
| contAlmcen | 13,00 | 306,01 | 0,00 | 13,00 | 13,00 | 5,67 |
| contMercado | 2,00 | 1.851,70 | 0,00 | 2,00 | 2,00 | 0,63 |

Como se puede ver tenemos que al aumentar las capacidades la cantidad de producto elaborado al final no cambia mucho el único cambio visible es la cantidad procesada de entidades por las primeras áreas. Mientras que en el área de fermentado tenemos que el total es de 96 y en su siguiente área es de 93 lo cual no cambia mucho.

- **Tabla de comparación:**

| | Prueba 1 | Prueba 2 | Prueba 3 |
|--------------------|-------------------|-------------------|-------------------|
| Área de fermentado | 96 litros mosto | 96 litros mosto | 96 litros mosto |
| Área de maduración | 18 litros alcohol | 93 litros alcohol | 93 litros alcohol |

Como se puede ver por la tabla y las pruebas hechas para poder aumentar la producción hay que reducir los litros de mosto utilizado en la mezcla con la levadura para poder producir más y también podría aumentarse la capacidad del horno.

Java:

Para el modelo de java al estar un poco mas limitados en temas de introducción de parámetro se experimentará con las cantidades de ciertas entidades que participan en mezclas a lo largo del proceso como la cantidad de lúpulo para formar mosto, la cantidad de mosto necesaria para elaborar alcohol y la cantidad de botellas de cerveza necesaria para crear una caja de cerveza.

- **Prueba 1:** Cambiando la cantidad de mosto utilizado en el proceso de fermentación a una cantidad de 5 por litros.

| Descripción | Valor |
|---|-------|
| Arribos | |
| Arribo de cebada | 122 |
| Arribo fallido de cebada | 45 |
| Arribo de lupulo | 392 |
| Arribo fallido de lupulo | 20 |
| Arribo de levadura | 47 |
| Arribo fallido de levadura | 166 |
| Arribo de caja vacia | 32 |
| Arribo fallido de caja vacia | 108 |
| Cantidades procesadas por cada locacion | |
| En malteado | 119 |
| En secado | 116 |
| En molienda | 113 |
| En maserado | 111 |
| En filtrado | 108 |
| En cocción | 98 |
| En enfriado | 96 |
| En fermentado | 95 |
| En maduración | 18 |
| En inspección | 18 |
| Rechazados | 2 |

Vemos que en el área de fermentación al requerir una menor cantidad de mosto para realizar las mezclas necesarias vemos que se logra producir una mayor cantidad de alcohol donde en fermentado hay 95 litros procesado y a maduración entraron 18.

- **Prueba 2:** Cambiando la cantidad de mosto utilizado en el proceso de fermentación a una cantidad de 8 por litros.

| Descripción | Valor |
|---|-------|
| Arribos | |
| Arribo de cebada | 122 |
| Arribo fallido de cebada | 45 |
| Arribo de lupulo | 392 |
| Arribo fallido de lupulo | 20 |
| Arribo de levadura | 32 |
| Arribo fallido de levadura | 181 |
| Arribo de caja vacia | 31 |
| Arribo fallido de caja vacia | 109 |
| Cantidades procesadas por cada locacion | |
| En malteado | 119 |
| En secado | 116 |
| En molienda | 113 |
| En maserado | 111 |
| En filtrado | 108 |
| En cocción | 98 |
| En enfriado | 96 |
| En fermentado | 95 |
| En maduración | 11 |
| En inspección | 11 |
| Rechazados | 1 |
| En embotellado | 10 |

Vemos que si volvemos a aumentar la cantidad de mosto utilizado a 8 nos da una menor cantidad de alcohol en este cado solo se produjo 11 litros de alcohol y se procesaron 95 litros de alcohol en el área de fermentación.

- Prueba 3: Probando con una cantidad de 1 litro de mosto por mezcla para realizar el alcohol.

| Descripcion | valor |
|---|-------|
| Arribos | |
| Arribo de cebada | 122 |
| Arribo fallido de cebada | 45 |
| Arribo de lupulo | 392 |
| Arribo fallido de lupulo | 20 |
| Arribo de levadura | 199 |
| Arribo fallido de levadura | 14 |
| Arribo de caja vacia | 42 |
| Arribo fallido de caja vacia | 98 |
| Cantidades procesadas por cada locacion | |
| En malteado | 119 |
| En secado | 116 |
| En molienda | 113 |
| En maserado | 111 |
| En filtrado | 108 |
| En cocción | 98 |
| En enfriado | 96 |
| En fermentado | 95 |
| En maduración | 89 |

Como se puede ver tenemos que cuando solo se pone un mosto para la mezcla la cantidad producida es mucho mayor debido a que no tiene que esperar el tiempo de producción de 10 mostos vemos que en fermentado tenemos 95 litros de mosto y en madurado salió un total de 89 litros de alcohol.

- **Tabla de comparación:**

| | Prueba 1 (5 lt) | Prueba 2 (8 lt) | Prueba 3 (1 lt) |
|--------------------|-------------------|-------------------|-------------------|
| Área de fermentado | 95 litros mosto | 95 litros mosto | 95 litros mosto |
| Área de maduración | 18 litros alcohol | 11 litros alcohol | 89 litros alcohol |

Como se puede ver por la tabla y las pruebas hechas para poder aumentar la producción de alcohol se debe utilizar menos mosto en la mezcla esto permitirá maximizar la producción ya que como se vio esa es la estación que limita la productividad de la cerveza.

Interpretación:

Pro Model:

Para interpretar los datos vemos primero el numero de entidades procesadas en cada área que son las siguientes:

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 124,00 | 33,87 | 0,00 | 124,00 | 124,00 | 68,02 |
| contMalteado | 121,00 | 34,56 | 0,00 | 121,00 | 121,00 | 65,68 |
| contSecado | 118,00 | 35,44 | 0,00 | 118,00 | 118,00 | 62,72 |
| contMolieda | 115,00 | 36,36 | 0,00 | 115,00 | 115,00 | 59,81 |
| contMaserado | 113,00 | 36,99 | 0,00 | 113,00 | 113,00 | 57,89 |
| contFiltrado | 110,00 | 38,00 | 0,00 | 110,00 | 110,00 | 55,05 |
| contCoccion | 108,00 | 38,71 | 0,00 | 108,00 | 108,00 | 53,51 |
| contSiloLupulo | 400,00 | 10,50 | 0,00 | 400,00 | 400,00 | 190,50 |
| contEnfriado | 98,00 | 42,66 | 0,00 | 98,00 | 98,00 | 45,67 |
| contSiloLevadura | 28,00 | 141,43 | 0,00 | 28,00 | 28,00 | 17,71 |
| contFermentado | 96,00 | 43,34 | 0,00 | 96,00 | 96,00 | 44,02 |
| contMaduracion | 9,00 | 448,99 | 0,00 | 9,00 | 9,00 | 3,77 |
| contInspeccion | 9,00 | 458,99 | 0,00 | 9,00 | 9,00 | 3,58 |
| contRechasada | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contEnbotellado | 9,00 | 462,32 | 0,00 | 9,00 | 9,00 | 3,51 |
| contEtiquetado | 9,00 | 462,99 | 0,00 | 9,00 | 9,00 | 3,50 |
| contEmpacado | 9,00 | 463,10 | 0,00 | 9,00 | 9,00 | 3,50 |
| contAlmacenCajas | 31,00 | 95,81 | 0,00 | 31,00 | 31,00 | 27,19 |
| contAlmcen | 1,00 | 2.978,12 | 0,00 | 1,00 | 1,00 | 0,29 |
| contMercado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |

Como se puede ver en la columna cantidad actual tenemos que en los silos de cebada se procesaron 124 kilogramos de grano, en el de malteado se procesaron 121, en secado 118, en molienda 115, en macerado 113, en filtrado 110, en cocción 108 litros, en silo de lupulo se procesaron 400 kilogramos de lupulo, en enfriado se procesaron 98 litros, en el silo de levadura se procesaron 28 kilogramos, en fermentado 96 litros, en maduración 9 litros, en inspección 9 litros, no existe rechazados, en embotellado, etiquetado, empacado se procesaron 9 botellas, en el almacén de cajas se procesaron 31 cajas en almacén solo 1 y en mercado 0.

El porcentaje de utilización de las áreas son:

| Nombre | Capacidad | Total Entradas | Contenido Promedio | Contenido Máximo | Contenido Actual | % Utilización |
|------------------|------------|----------------|--------------------|------------------|------------------|---------------|
| Silo cebada | 3,00 | 124,00 | 2,34 | 3,00 | 3,00 | 77,94 |
| Malteado | 3,00 | 121,00 | 2,96 | 3,00 | 3,00 | 98,54 |
| Secado | 3,00 | 118,00 | 2,90 | 3,00 | 3,00 | 96,81 |
| Molienda | 2,00 | 115,00 | 1,92 | 2,00 | 2,00 | 95,86 |
| Maserado | 3,00 | 113,00 | 2,84 | 3,00 | 3,00 | 94,58 |
| Filtrado | 2,00 | 110,00 | 1,54 | 2,00 | 2,00 | 77,23 |
| Coccion | 10,00 | 108,00 | 7,84 | 10,00 | 10,00 | 78,35 |
| Silo lupulo | 10,00 | 400,00 | 0,83 | 10,00 | 0,00 | 8,31 |
| Enfriado | 10,00 | 98,00 | 1,65 | 4,00 | 2,00 | 16,52 |
| Silo levadura | 10,00 | 28,00 | 9,66 | 10,00 | 10,00 | 96,59 |
| Fermentado | 10,00 | 96,00 | 6,33 | 10,00 | 6,00 | 63,25 |
| Maduracion | 10,00 | 9,00 | 0,19 | 1,00 | 0,00 | 1,93 |
| Inspeccion | 3,00 | 9,00 | 0,06 | 1,00 | 0,00 | 2,14 |
| Enbotellado | 6,00 | 9,00 | 0,01 | 1,00 | 0,00 | 0,21 |
| Etiquetado | 6,00 | 9,00 | 0,00 | 1,00 | 0,00 | 0,04 |
| Almacen de cajas | 30,00 | 31,00 | 26,89 | 30,00 | 30,00 | 89,64 |
| Empacado | 6,00 | 9,00 | 1,75 | 6,00 | 3,00 | 29,19 |
| Almacenaje | 6,00 | 1,00 | 0,29 | 1,00 | 1,00 | 4,85 |
| Mercado | 999.999,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |

Como se puede ver tenemos que la mayoría de las primeras locaciones tienen el porcentaje de utilización alto, pero pasando del área de fermentado vemos que las áreas no se utilizan debido a que tienen que esperar a que el área de fermentado bote el alcohol para poder seguir con su operación.

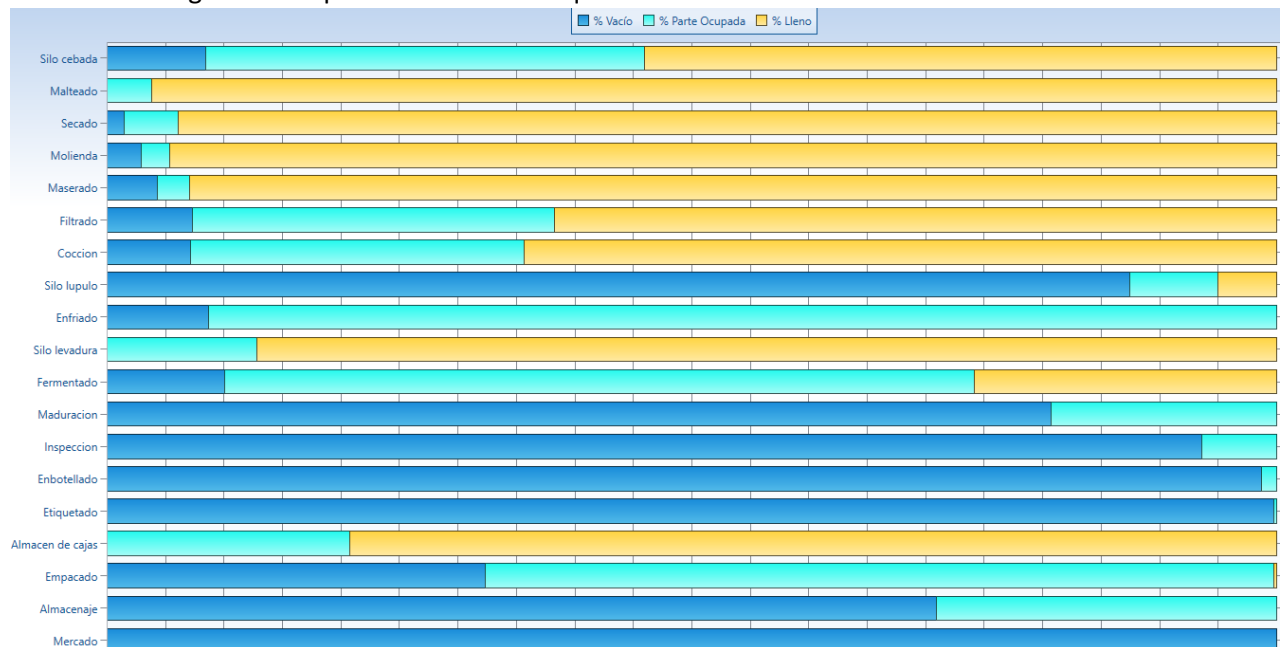
Por parte de las entidades tenemos que:

| Nombre | Total Salidas | Cantidad actual En Sistema | Tiempo En Sistema Promedio (Min) | Tiempo Esperando Promedio (Min) | Tiempo de Bloqueo Promedio (Min) |
|--------------------|---------------|----------------------------|----------------------------------|---------------------------------|----------------------------------|
| Grano cebada | 0,00 | 26,00 | 0,00 | 0,00 | 0,00 |
| Lupulo | 399,00 | 1,00 | 9,03 | 8,69 | 0,00 |
| Mosto | 90,00 | 8,00 | 1.043,34 | 414,48 | 207,19 |
| Levadura | 18,00 | 10,00 | 1.670,71 | 1.670,28 | 0,00 |
| Alcohol | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| Botella de cerveza | 6,00 | 3,00 | 1.247,64 | 1.000,64 | 0,00 |
| Caja vacia | 1,00 | 30,00 | 2.967,92 | 2.967,92 | 0,00 |
| Caja con cerveza | 0,00 | 1,00 | 0,00 | 0,00 | 0,00 |

El total de salidas de cebada es 0 debido a que se convierte en alcohol, el total de salidas es de 399 debido a que se utiliza en el horno, el total de salidas de mosto es de 90 debido a que se utiliza en fermentado, el de la levadura es de 18 debido a que se utiliza en fermentado y alcohol es 0 debido a que se transforma en botella de cerveza, las salidas de botella es de 6 debido a que se convierten en un paquete, el de la caja es 1 por que se utilizan para el paquete y el de caja de cerveza es de 0 debido a que deben esperar a ser 6 para poder ir al mercado.

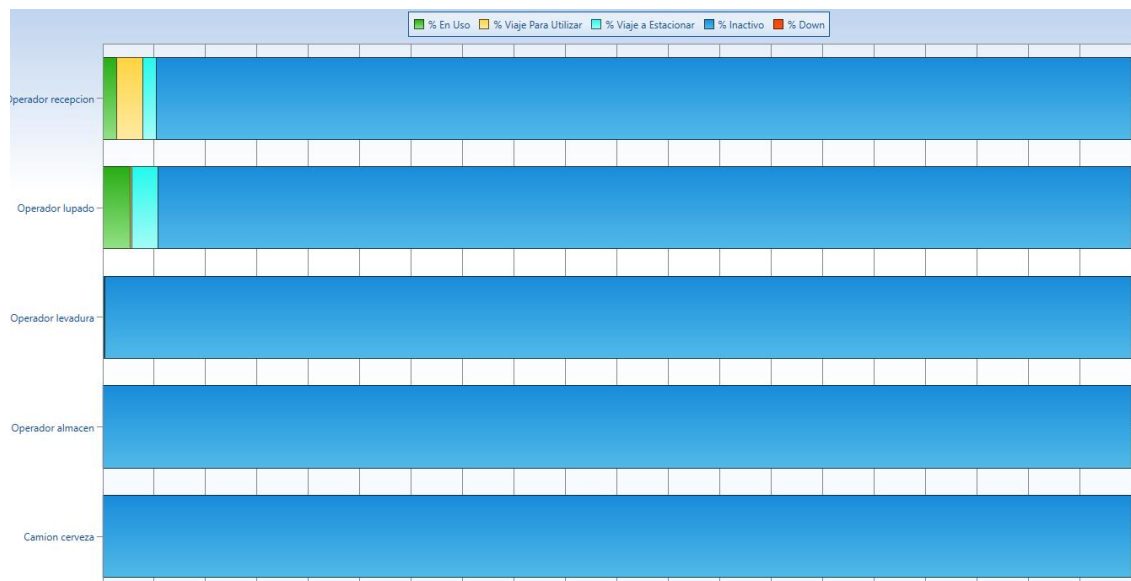
Graficas Pro Model:

Analizando las gráficas de pro model tenemos que:



Viendo tenemos que las primeras locaciones andan la mayor parte del tiempo llenas debido a que tienen que procesar el grano al inicio, también vemos que las locaciones después de fermentado andan la mayor parte del tiempo vacías que coincide con lo visto anterior mente en los porcentajes de utilización también vemos que el ciclo de lupulo anda vacío una gran parte del tiempo debido a que por cada kilogramo tenemos que utilizar 4 de lupulo y en los arribos de

las otras áreas tenemos que andan la mayor parte del tiempo llenas debido a que no se utilizan mucho.



Para los operadores tenemos que la mayor parte del tiempo igual no se están utilizando en si solo se utilizan cuando se requiere mezclar las entidades en las diferentes áreas.

Java:

Para interpretar los datos vemos primero el número de entidades procesadas en cada área que son las siguientes:

| Cantidades procesadas por cada locacion | |
|---|-----|
| En malteado | 119 |
| En secado | 116 |
| En molienda | 113 |
| En maserado | 111 |
| En filtrado | 108 |
| En cocción | 98 |
| En enfriado | 96 |
| En fermentado | 95 |
| En maduración | 9 |
| En inspección | 9 |
| Rechazados | 1 |
| En embotellado | 7 |
| En etiquetado | 7 |
| En empacado | 7 |
| En almacen | 1 |
| En mercado | 0 |
| Entidades en proceso | |

Vemos que las cantidades procesadas por cada área de trabajo son las siguientes en malteado se proceso 119 kg de grano, en secado se proceso 116 kg de grano, en el área de molienda se proceso 113 kg de grano, en el área de macerado se proceso 111 kg de grano, en el área de filtrado se proceso 108 kg de grano, en el área de cocción se proceso 98 kg de grano, en el área de enfriado se procesaron 10 litros de mosto, en el área de fermentado se procesaron 95 litros de mosto, en el área de maduración se procesaron 9 litros de alcohol, en el área de inspección se procesaron 9 litros de alcohol de los cuales se rechazo 1, en embotellado se prosearcharon 7 litros de alcohol, en el área de etiquetado se procesaron 7 botellas, en el área de empacado se procesaron 7 botellas, en el almacén se procesaron 1 caja de cerveza y en mercado nada.

Por parte de los arribos tenemos los siguientes datos.

| Descripción | Valor |
|------------------------------|-------|
| Arribos | |
| Arribo de cebada | 122 |
| Arribo fallido de cebada | 45 |
| Arribo de lupulo | 392 |
| Arribo fallido de lupulo | 20 |
| Arribo de levadura | 28 |
| Arribo fallido de levadura | 185 |
| Arribo de caja vacia | 31 |
| Arribo fallido de caja vacia | 109 |

Vemos que al silo de cebada arribaron 122 kg de grano y fallaron un total de 45, al silo de lúpulo llegaron un total de 392 kg de lúpulo y fallaron 20, mientras que a silo de levadura llegaron un total de 28 kg de levadura y fallaron 185 kg en llegar y finalmente al almacén de cajas vemos que llegaron 31 cajas y fallaron 109 cajas en llegar.

Mientras que para las entidades todavía en proceso están lo siguiente.

| | |
|-----------------------------|----|
| Entidades en proceso | |
| En silo de cebada | 3 |
| En Malteado | 3 |
| En secado | 3 |
| En molienda | 2 |
| En maserado | 3 |
| En filtrado | 2 |
| En silo de lupulo | 0 |
| En cocción | 10 |
| En enfriado | 1 |
| En silo de levadura | 10 |
| En fermentación | 5 |
| En maduración | 0 |
| En inspección | 1 |
| En embotellado | 0 |
| En etiquetado | 0 |
| En almacen de cajas | 30 |
| En empacado | 1 |
| En almacen | 1 |
| En mercado | 0 |

Vemos que en el silo de secada hay 3 kg de grano, en el de malteado hay 3 en el de secado hay 3 en el de molienda hay 3, en el de macerado hay 3, en el de filtrado hay 2, en el silo de lúpulo este vacío, el área de cocción esta llena con 10, en enfriado hay 1, en silo de levadura hay 10 kg de levadura, en fermentación existen 5 litros de mosto en este momento en inspección hay 1 y en almacén de cajas hay 30 cajas actualmente.

Mientras vemos que el total de salidas de las entidades y el tiempo promedio que pasan en sistema son.

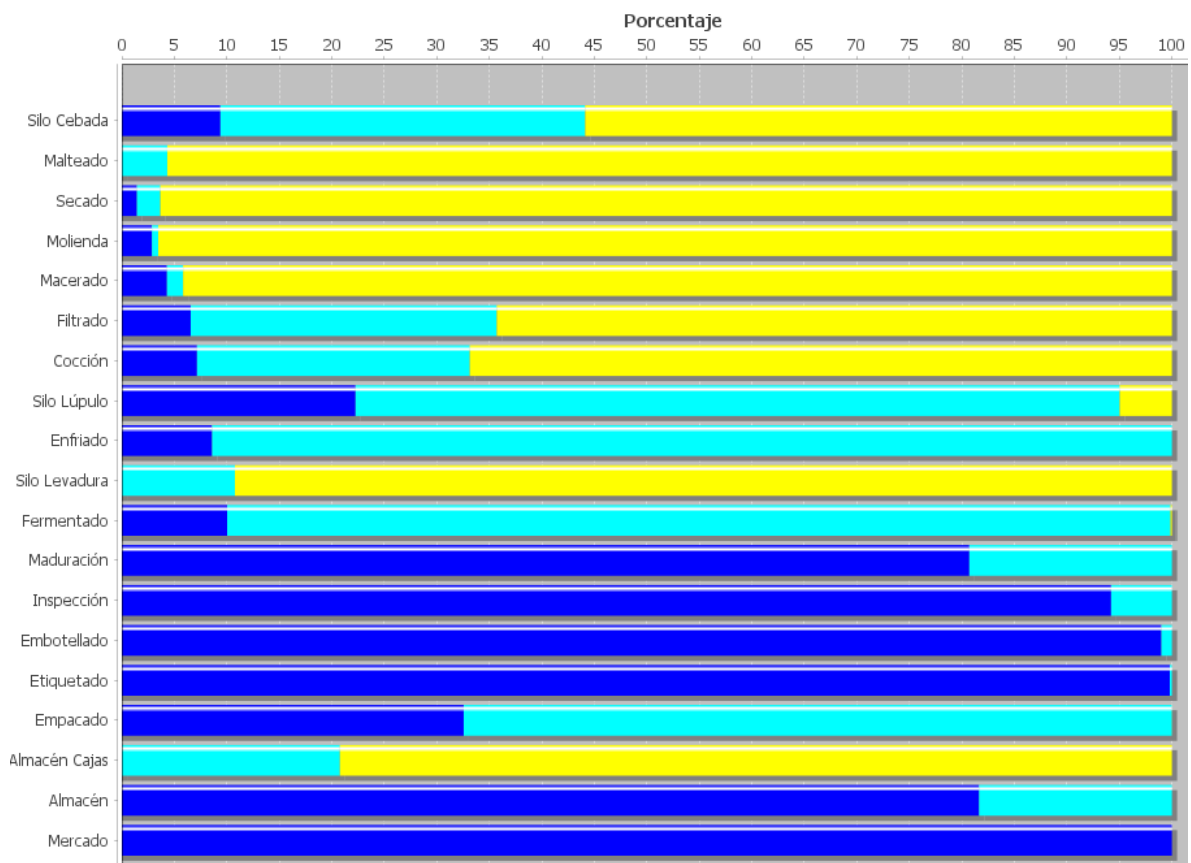
| | |
|--|--------|
| Total salidas | |
| Salidas de grano de cebada | 0 |
| Salidas de lupulo | 392 |
| Salidas de Mosto | 90 |
| Salidas de levadura | 18 |
| Salidas de alcohol | 0 |
| Salidas de botellas de cerveza | 6 |
| Salidas de caja vacía | 1 |
| Salidas de cajas de cerveza | 0 |
| Tiempo promedio en sistema en minutos | |
| Tiempo promedio en sistema de grano d... | 0.0 |
| Tiempo promedio en sistema de lupulo | 8.9 |
| Tiempo promedio en sistema de Mosto | 1041.1 |
| Tiempo promedio en sistema de levadura | 1670.0 |
| Tiempo promedio en sistema de alcohol | 0.0 |
| Tiempo promedio en sistema de botella... | 1245.3 |
| Tiempo promedio en sistema de caja va... | 2967.1 |
| Tiempo promedio en sistema de cajas d... | 0.0 |

Tenemos que la total salida de grano es de 0 por que este se convierte en alcohol, en lúpulo el total de salidas es de 392 por que se usaron en las mezclas, en mosto de igual forma salieron 90, y de levadura salió 18 por que se utilizaron para crear alcohol y de igual forma las botellas de cerveza y las cajas se utilizaron para crear cajas de cerveza.

Vemos que los tiempos promedios son de 1041.1 minutos para el mosto, 1670 para la levadura, 1245.3 para las botellas y 2967.1 para las cajas vacías.

Graficas Java:

Para la parte grafica vemos el grado en el que las locaciones están ocupadas vacías o están llenas.



Vemos que la mayoría de las áreas se encontraron llenas la mayor parte del tiempo, eso las primeras áreas debido a que estas son las que mas se encuentran en operación y también vemos que las áreas de abajo es decir las áreas después de fermentado casi nunca están llenas debido a que esperan el alcohol para trabajar lo que refleja el comportamiento planteado en el problema.

Documentación:

Archivo de texto Pro Model:

El archivo de texto generado por el modelo de simulación de Pro Model es el siguiente.

```

*                               Listado del modelo formateado:
*                               D:\UMSS\Semestre 2-2024\TSS\Proyecto3\proyecto3.mod
*
*****

```

```

Unidades de Tiempo:           Minutos
Unidades de Distancia:       Pies

```

```

*****
*                               Locaciones
*****

```

| Nombre | Cap | Unidades | Estadist | Reglas | Costos |
|------------------|-----|----------|------------------|---------------|--------|
| Silo_cebada | 3 | 1 | Series de tiempo | Más Tiempo, , | |
| Malteado | 3 | 1 | Series de tiempo | Más Tiempo, , | |
| Secado | 3 | 1 | Series de tiempo | Más Tiempo, , | |
| Molienda | 2 | 1 | Series de tiempo | Más Tiempo, , | |
| Maserado | 3 | 1 | Series de tiempo | Más Tiempo, , | |
| Filtrado | 2 | 1 | Series de tiempo | Más Tiempo, , | |
| Coccion | 10 | 1 | Series de tiempo | Más Tiempo, , | |
| Silo_lupulo | 10 | 1 | Series de tiempo | Más Tiempo, , | |
| Enfriado | 10 | 1 | Series de tiempo | Más Tiempo, , | |
| Silo_levadura | 10 | 1 | Series de tiempo | Más Tiempo, , | |
| Fermentado | 10 | 1 | Series de tiempo | Más Tiempo, , | |
| Maduracion | 10 | 1 | Series de tiempo | Más Tiempo, , | |
| Inspeccion | 3 | 1 | Series de tiempo | Más Tiempo, , | |
| Enbotellado | 6 | 1 | Series de tiempo | Más Tiempo, , | |
| Etiquetado | 6 | 1 | Series de tiempo | Más Tiempo, , | |
| Almacen_de_cajas | 30 | 1 | Series de tiempo | Más Tiempo, , | |
| Empacado | 6 | 1 | Series de tiempo | Más Tiempo, , | |
| Almacenaje | 6 | 1 | Series de tiempo | Más Tiempo, , | |
| Mercado | INF | 1 | Series de tiempo | Más Tiempo, , | |

```

*                               Entidades
*****

```

| Nombre | Velocidad (Ppm) | Estadist | Costos |
|--------------------|-----------------|------------------|--------|
| Grano_cebada | 150 | Series de tiempo | |
| Lupulo | 150 | Series de tiempo | |
| Mosto | 150 | Series de tiempo | |
| Levadura | 150 | Series de tiempo | |
| Alcohol | 150 | Series de tiempo | |
| Botella_de_cerveza | 150 | Series de tiempo | |
| Caja_vacia | 150 | Series de tiempo | |
| Caja_con_cerveza | 150 | Series de tiempo | |

```

*****
*                               Redes de Ruta
*****

```

| Nombre | Tipo | T/V | Desde | Hasta | BI | Distancia/Tiempo | Factor de Velocidad |
|-------------|------------|-----------------------|-------|-------|----|------------------|---------------------|
| trabajador1 | Sobrepasar | Velocidad & Distancia | N1 | N2 | Bi | 18.53 | 1 |
| | | | N2 | N3 | Bi | 21.75 | 1 |
| | | | N3 | N4 | Bi | 17.11 | 1 |
| trabajador2 | Sobrepasar | Velocidad & Distancia | N1 | N2 | Bi | 28.43 | 1 |
| trabajador3 | Sobrepasar | Velocidad & Distancia | N1 | N2 | Bi | 21.35 | 1 |
| trabajador4 | Sobrepasar | Velocidad & Distancia | N1 | N2 | Bi | 19.67 | 1 |
| rutaCamion | Sobrepasar | Velocidad & Distancia | N1 | N2 | Bi | 42.48 | 1 |

| * Interfaces * | | | | | | | |
|--------------------|---------------|------------------------------|-------------|--------------|-------------|--|--|
| ***** | | | | | | | |
| Red | Nodo | Locación | | | | | |
| ----- | | | | | | | |
| trabajador1 | N1 | Silo_lupulo | | | | | |
| | N2 | Molienda | | | | | |
| | N3 | Secado | | | | | |
| | N4 | Malteado | | | | | |
| trabajador2 | N1 | Silo_lupulo | | | | | |
| | N2 | Coccion | | | | | |
| trabajador3 | N1 | Silo_levadura | | | | | |
| | N2 | Fermentado | | | | | |
| trabajador4 | N1 | Empacado | | | | | |
| | N2 | Almacenaje | | | | | |
| rutaCamion | N1 | Almacenaje | | | | | |
| | N2 | Mercado | | | | | |
| ***** | | | | | | | |
| * Recursos * | | | | | | | |
| ***** | | | | | | | |
| | | Rec | | Ent | | | |
| Nombre | Unidades | Estadíst | Buscar | Buscar | Ruta | Movimiento | Costos |
| ----- | | | | | | | |
| Operador_recepcion | 1 | Por Unidad | Ninguna | Más Tiempo | trabajador1 | Vacío: 150 Ppm Home: N1 (Regresar) | |
| Operador_lupado | 1 | Por Unidad | Más Cercano | Más Tiempo | trabajador2 | Vacío: 100 Ppm Home: N1 (Regresar) | |
| Operador_levadura | 1 | Por Unidad | Más Cercano | Más Tiempo | trabajador3 | Vacío: 100 Ppm Home: N1 (Regresar) | |
| ----- | | | | | | | |
| * Procesamiento * | | | | | | | |
| ***** | | | | | | | |
| | | Proceso | | Enrutamiento | | | |
| Entidad | Locación | Operación | Blk | Salida | Destino | Regla | Lógica de Movimiento |
| ----- | | | | | | | |
| Grano_cebada | Silo_cebada | contSiloCebada = Entries() | 1 | Grano_cebada | Malteado | FIRST 1 | Inc contMalteado, 1 |
| Grano_cebada | Malteado | //contMalteado = Entries() | 1 | Grano_cebada | Secado | FIRST 1 | Move With Operador_recepcion Then Free |
| Grano_cebada | Secado | contSecado = Entries() | 1 | Grano_cebada | Molienda | FIRST 1 | Move With Operador_recepcion Then Free |
| Grano_cebada | Molienda | contMolienda = Entries() | 1 | Grano_cebada | Maserado | FIRST 1 | |
| Grano_cebada | Maserado | contMaserado = Entries() | 1 | Grano_cebada | Filtrado | FIRST 1 | |
| Grano_cebada | Filtrado | contFiltrado = Entries() | 1 | Grano_cebada | Coccion | FIRST 1 | |
| Grano_cebada | Coccion | contCoccion = Entries() | 1 | Grano_cebada | Enfriado | FIRST 1 | |
| Lupulo | Silo_lupulo | Join LupuloCoccion Lupulo | 1 | Mosto | Enfriado | FIRST 1 | |
| Lupulo | Silo_lupulo | contSiloLupulo = Entries() | 1 | Lupulo | Coccion | Join 1 | Move With Operador_lupado Then Free |
| Mosto | Enfriado | contEnfriado = Entries() | 1 | Mosto | Fermentado | FIRST 1 | |
| Levadura | Silo_levadura | contSiloLevadura = Entries() | 1 | Levadura | Fermentado | Join 1 | Move With Operador_levadura Then Free |
| Mosto | Fermentado | contFermentado = Entries() | 1 | Alcohol | Maduracion | FIRST 1 | |
| Alcohol | Maduracion | Combine MesclaMosto | 1 | Alcohol | Inspeccion | FIRST 1 | |
| Alcohol | Maduracion | Join 2 Levadura | 1 | Alcohol | Inspeccion | FIRST 1 | |
| Alcohol | Maduracion | Wait 2 hr | 1 | Alcohol | Inspeccion | FIRST 1 | |
| Alcohol | Maduracion | contMaduracion = Entries() | 1 | Alcohol | Inspeccion | FIRST 1 | |
| Alcohol | Maduracion | Wait 1.5 hr | 1 | Alcohol | Inspeccion | FIRST 1 | |

```

Alcohol      Maduracion      contMaduracion = Entries()
Wait 1.5 hr      1      Alcohol      Inspeccion FIRST 1
Alcohol      Inspeccion      contInspeccion = Entries()
Wait 0.5 hr      1      Alcohol      Enbotellado 0.9 1
Alcohol      Enbotellado      contEnbotellado = Entries()
Wait 6 min      1      Botella_de_cerveza Etiquetado FIRST 1
Botella_de_cerveza Etiquetado contEtiquetado = Entries()
Wait 1 min      1      Botella_de_cerveza Empacado FIRST 1
Botella_de_cerveza Empacado   contEmpacado = Entries()
Combine BotellasCaja
Join 1 Caja_vacia
Wait 10 min      1      Caja_con_cerveza Almacenaje FIRST 1 Move With Operador_almacen Then Free

Caja_vacia      Almacen_de_cajas contAlmacenCajas = Entries()
1      Caja_vacia      Empacado      Join 1
Caja_con_cerveza Almacenaje      contAlmcen = Entries()
Combine 6
Wait 5 min      1      Caja_con_cerveza Mercado FIRST 1 Move With Camion_cerveza Then Free

Caja_con_cerveza Mercado      contMercado = Entries()
1      Caja_con_cerveza EXIT FIRST 1

```

```

*****
*                               *
*                               *
*****

```

| Entidad | Locación | Cant. por Arribo | Primera Vez | Ocurrencias | Frecuencia Lógica |
|--------------|------------------|------------------|-------------|-------------|-------------------|
| Grano_cebada | Silo_cebada | 1 | 0 | INF | 25 min |
| Lupulo | Silo_lupulo | 1 | 0 | INF | 10 min |
| Levadura | Silo_levadura | 1 | 0 | INF | 20 min |
| Caja_vacia | Almacen_de_cajas | 1 | 0 | INF | 30 min |

```

*****
*                               *
*                               *
*****

```

Variables (global)

| ID | Tipo | Valor Inicial | Estadist |
|------------------|---------|---------------|------------------|
| contSiloCebada | Integer | 0 | Series de tiempo |
| contMalteado | Integer | 0 | Series de tiempo |
| contSecado | Integer | 0 | Series de tiempo |
| contMolieda | Integer | 0 | Series de tiempo |
| contMaserado | Integer | 0 | Series de tiempo |
| contFiltrado | Integer | 0 | Series de tiempo |
| contCoccion | Integer | 0 | Series de tiempo |
| contSiloLupulo | Integer | 0 | Series de tiempo |
| contEnfriado | Integer | 0 | Series de tiempo |
| contSiloLevadura | Integer | 0 | Series de tiempo |
| contFermentado | Integer | 0 | Series de tiempo |
| contMaduracion | Integer | 0 | Series de tiempo |
| contInspeccion | Integer | 0 | Series de tiempo |
| contRechasada | Integer | 0 | Series de tiempo |
| contEnbotellado | Integer | 0 | Series de tiempo |
| contEtiquetado | Integer | 0 | Series de tiempo |
| contEmpacado | Integer | 0 | Series de tiempo |
| contAlmacenCajas | Integer | 0 | Series de tiempo |
| contAlmcen | Integer | 0 | Series de tiempo |
| contMercado | Integer | 0 | Series de tiempo |

```

*****
*                               *
*                               *
*****

```

Macros

| ID | Texto |
|---------------|-------|
| MesclaMosto | 10 |
| LupuloCoccion | 4 |
| BotellasCaja | 6 |

Código de java:

Interfaz de entrada:

```
package com.mycompany.proyecto3;

public class Proyecto3 extends javax.swing.JFrame {

    public Proyecto3() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    Generated Code

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        EjerUI ejercicio = new EjerUI();
        ejercicio.setVisible(true);
    }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Proyecto3().setVisible(true);
            }
        });
    }
}
```

Se encarga de llamar a la interfaz principal que es donde se pondrán los datos para poder realizar la simulación dicha interfaz se llama EjerUI.

Interfaz de datos EjerUI:

```
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JScrollPane;
import javax.swing.JTabbedPane;
import javax.swing.JTextArea;
import javax.swing.JTextField;

public class EjerUI extends javax.swing.JFrame {

    private JTextArea textAreaA;
    private JTextArea textAreaB;
    private JTextArea textAreaC;

    public EjerUI() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    private void initComponents() { [...122 lines] }

    private void simularEjercicioAction(java.awt.event.ActionEvent evt, JTextField simulacionHoras, JTextField replicas, JTextField delay, JTextField capacidad, JTextField litros, JTextField botellas) {
        try {
            int duracionHoras = Integer.parseInt(simulacionHoras.getText());
            int numReplicas = Integer.parseInt(replicas.getText());
            int delaySegundos = Integer.parseInt(delay.getText());
            int capacidad = Integer.parseInt(capacidadLupulo.getText()); // Ajustado
            int litros = Integer.parseInt(litrosMosto.getText());
            int botellas = Integer.parseInt(botellasCaja.getText());

            Ejecucion hilo = new Ejecucion(textAreaA, textAreaB, duracionHoras, numReplicas, delaySegundos, capacidad, litros, botellas);
            hilo.start();
        } catch (NumberFormatException e) {
            JOptionPane.showMessageDialog(this, "Por favor, introduce valores válidos.", "Error", JOptionPane.ERROR_MESSAGE);
        }
    }
}
```

Se encarga de mostrar la interfaz donde se insertan los datos de simulación en este caso dicha interfaz se encarga de recibir los datos de duración de la simulación, el numero de replicas, el delay de las animaciones, la capacidad de lúpulo para mezclar en cocción, la cantidad de litros de mosto para realizar las mezclas con la levadura y la cantidad de botellas que tendrá cada paquete.

Luego manda a llamar a la simulación y al hilo respectivo para dicha simulación este se llama Ejecución.


```

4   import java.awt.Dimension;
5   import java.awt.Graphics;
6   import java.awt.image.BufferedImage;
7   import java.io.IOException;
8   import java.math.BigDecimal;
9   import java.math.RoundingMode;
10  import java.util.ArrayList;
11  import java.util.List;
12  import javax.imageio.ImageIO;
13  import javax.swing.JFrame;
14  import javax.swing.JPanel;
15  import javax.swing.JTextArea;
16  import javax.swing.SwingUtilities;
17  import javax.swing.Timer;
18  import javax.swing.*;
19  import javax.swing.event.ChangeEvent;
20  import javax.swing.event.ChangeListener;
21  import java.util.Random;
22
23
24  public class Ejecucion extends Thread {
25      private JTextArea textAreaA;
26      private JTextArea textAreaB;
27      private int duracion;
28      private int replicas;
29      private volatile int delay; // Volatile para garantizar cambios en tiempo real
30      private int capacidadCoccion;
31      private int litrosMosto;
32      private int botellaPaquete;
33      private int contSiloCebada = 0; private int contSiloLupulo=0;
34      private int contSiloLevadura = 0; private int contAlmacenCaja = 0;
35      private int contSiloCebadaF = 0; private int contSiloLupuloF=0;
36      private int contSiloLevaduraF = 0; private int contAlmacenCajaF = 0;
37

```

Primero realizamos las exportaciones correspondientes de las librerías que se utilizarán para el desarrollo de la simulación y creamos las variables que contendrán los datos que mandamos de la interfaz de datos.

```

38  // Otras variables (sin cambios)...
39  private SimulacionPanel simPanel;
40  private JSlider delaySlider;
41  public Ejecucion(JTextArea textAreaA, JTextArea textAreaB, int duracion, int replicas, int delay, int capacidadCoccion,
42      int litrosMosto, int botellaPaquete) {
43      this.textAreaA = textAreaA;
44      this.textAreaB = textAreaB;
45      this.duracion = duracion;
46      this.replicas = replicas;
47      this.delay = delay;
48      this.capacidadCoccion = capacidadCoccion;
49      this.litrosMosto = litrosMosto;
50      this.botellaPaquete = botellaPaquete;
51
52      initGrafico(); // Inicializar gráfica y slider
53  }

```

Tenemos el constructor de la clase para poder iniciar el objeto y realizar su ejecución e inicializar las variables creadas anteriormente, también manda a iniciar el grafico de la simulación.

```

55 private void initGrafico() {
56     JFrame frame = new JFrame("Simulación con Control de Delay");
57     frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
58     frame.setSize(1300, 800);
59
60     simPanel = new SimulacionPanel();
61     simPanel.setPreferredSize(new Dimension(900, 600));
62
63     // Crear el slider para controlar el delay
64     delaySlider = new JSlider(JSlider.HORIZONTAL, 0, 1000, delay);
65     delaySlider.setMajorTickSpacing(200);
66     delaySlider.setMinorTickSpacing(50);
67     delaySlider.setPaintTicks(true);
68     delaySlider.setPaintLabels(true);
69
70     // Listener para actualizar el delay en tiempo real
71     delaySlider.addChangeListener(new ChangeListener() {
72         @Override
73         public void stateChanged(ChangeEvent e) {
74             delay = delaySlider.getValue();
75         }
76     });
77
78     // Crear un panel para el slider y agregarlo
79     JPanel controlPanel = new JPanel();
80     controlPanel.add(new JLabel("Delay:"));
81     controlPanel.add(delaySlider);
82
83     // Agregar componentes al marco
84     frame.setLayout(new BorderLayout(frame.getContentPane(), BorderLayout.Y_AXIS));
85     frame.add(simPanel);
86     frame.add(controlPanel);
87
88     frame.setVisible(true);
89 }

```

Luego construimos el método que generara el grafico inicial donde se mostrara todas las acciones en este le damos el tamaño de la ventana el nombre y también creamos un scroll para poder controlar el delay en tiempo real en la simulación.

```

91 @Override
92 public void run() {
93     //Locacion
94     List<Entidad> siloCebada = new ArrayList<>();
95     List<Entidad> malteado = new ArrayList<>();
96     List<Entidad> secado = new ArrayList<>();
97     List<Entidad> molienda = new ArrayList<>();
98     List<Entidad> maserado = new ArrayList<>();
99     List<Entidad> filtrado = new ArrayList<>();
100     List<Entidad> coccion = new ArrayList<>();
101     List<Entidad> siloLupulo = new ArrayList<>();
102     List<Entidad> enfriado = new ArrayList<>();
103     List<Entidad> siloLevadura = new ArrayList<>();
104     List<Entidad> fermentado = new ArrayList<>();
105     List<Entidad> maduracion = new ArrayList<>();
106     List<Entidad> inspeccion = new ArrayList<>();
107     List<Entidad> embotellado = new ArrayList<>();
108     List<Entidad> etiquetado = new ArrayList<>();
109     List<Entidad> empacado = new ArrayList<>();
110     List<Entidad> almacenCajas = new ArrayList<>();
111     List<Entidad> almacen = new ArrayList<>();
112     List<Entidad> mercado = new ArrayList<>();
113
114     //Recursos
115     List<Entidad> operadorRecepcion = new ArrayList<>();
116     List<Entidad> trabajadorLupulo = new ArrayList<>();
117     List<Entidad> trabajadorLevadura = new ArrayList<>();
118     List<Entidad> trabajadorCajas = new ArrayList<>();
119     List<Entidad> camion = new ArrayList<>();
120
121     //Variables auxiliares
122     double arriboCebada = 0;
123     double arriboLupulo = 0;
124     double arriboLevadura = 0;
125     double arriboCaja = 0;
126
127     //Variables de resultados

```

Luego iniciamos la ejecución del proceso inicializando las variables que se usaran.

```

154
155 //Simular tiempo
156 for (int k = 1; k <= replicas; k++) {
157     for (int i = 0; i <= duracion; i++) { // Simular las horas
158         if (i != duracion) { // Si no es la última hora
159             BigDecimal j = new BigDecimal("0.00");
160             BigDecimal one = new BigDecimal("1.0");
161
162             while (j.compareTo(one) < 0) { // Mientras j < 1.0
163                 BigDecimal horaExacta = new BigDecimal(i).add(j).setScale(2, RoundingMode.HALF_UP);
164                 double tiempoActual = horaExacta.doubleValue();
165                 textAreaA.append("Hora " + tiempoActual + "\n");
166                 //Calculo de datos
167                 if (siloCebada.size() == 3) { contArealL++; }
168                 if (malteado.size() == 3) { contArea2L++; }
169                 if (secado.size() == 3) { contArea3L++; }
170                 if (molienda.size() == 2) { contArea4L++; }
171                 if (maserado.size() == 3) { contArea5L++; }
172                 if (filtrado.size() == 2) { contArea6L++; }
173                 if (coccion.size() == 10) { contArea7L++; }
174                 if (siloLupulo.size() == 10) { contArea8L++; }
175                 if (enfriado.size() == 10) { contArea9L++; }
176                 if (siloLevadura.size() == 10) { contAreal0L++; }
177                 if (fermentado.size() == 10) { contAreal1L++; }
178                 if (maduracion.size() == 10) { contAreal2L++; }
179                 if (inspeccion.size() == 3) { contAreal3L++; }
180                 if (embotellado.size() == 6) { contAreal4L++; }
181                 if (etiquetado.size() == 6) { contAreal5L++; }
182                 if (empacado.size() == 6) { contAreal6L++; }
183                 if (almacenCajas.size() == 30) { contAreal7L++; }
184                 if (almacen.size() == 6) { contAreal8L++; }
185                 if (mercado.size() == 100) { contAreal9L++; }
186
187                 if (siloCebada.isEmpty()) { contArealV++; }
188                 if (malteado.isEmpty()) { contArea2V++; }

```

Luego mediante ciclos for iniciamos el conteo del tiempo para poder realizar la simulación también se inicia el numero de replicas y el tiempo entre hora que trabajara con una exactitud de 2 decimales, también se realizar preguntas para saber el estado actual de las locaciones y guardar dichos datos.

```

208 //Arribos
209 if(tiempoActual == 0.00 || tiempoActual==arriboCebada){ //cebada
210     arriboCebada = arribo(siloCebada, 3, "cebada", 0.42, tiempoActual); //cada 25 minutos
211     simPanel.llegadaPieza("siloGrano");
212 }
213 if(tiempoActual == 0.00 || tiempoActual==arriboLupulo){ //Lupulo
214     arriboLupulo = arribo(siloLupulo, 10, "lupulo", 0.17, tiempoActual); //cada 10 minutos
215     simPanel.llegadaPieza("siloLupulo");
216 }
217 if(tiempoActual == 0.00 || tiempoActual==arriboLevadura){ //Levadura
218     arriboLevadura = arribo(siloLevadura, 10, "levadura", 0.33, tiempoActual); //cada 20 minutos
219     simPanel.llegadaPieza("siloLevadura");
220 }
221 if(tiempoActual == 0.00 || tiempoActual==arriboCaja){ //Cajas
222     arriboCaja = arribo(almacenCajas, 30, "caja vacia", 0.5, tiempoActual); //cada 30 minutos
223     simPanel.llegadaPieza("almacenCaja");
224 }
225 //Procesos
226 simPanel.actualizarEstado(
227     contSiloCebada, contSiloLupulo, contSiloLevadura, contAlmacenCaja,
228     contMalteado, contSecado, contMolienda, contMaserado, contFiltrado, contMosto,
229     contEnfriado, contAlcohol, contMaduracion, contIns,
230     contRechazado, contEmbotellado, contEtiquetado, contEmpacado,
231     contAlmacen, contMercado);
232 //Salida de almacen

```

Luego realizamos los arribos correspondientes a las diferentes locaciones preguntando si es la primera vez y después preguntando si es tiempo de un nuevo arribo.

Luego actualizamos los datos de la grafica para poder ver la cantidad de arribos ocurridos.

```

233 if(!almacen.isEmpty()){ //almacen no esta vacio
234     for(int al = 0; al < almacen.size(); al++){ //recorremos cada entidad
235         Entidad paquete = almacen.get(al); //guardamos la entidad
236         if(paquete.getTiempoL4() == tiempoActual){ //vemos si es tiempo de salida
237             almacen.remove(paquete); //eliminamos de almacen
238             simPanel.llegadaPieza("almacenVacio");
239             retrasoGrafica(delay);
240             textAreaA.append("Paquete se envio a mercado\n");
241             camion.add(paquete); //añadimos a camion
242             simPanel.llegadaPieza("camionLleno");
243             retrasoGrafica(delay);
244             simPanel.moverRecurso("camion", 970, 550);
245             camion.remove(paquete); //eliminamos de camion
246             simPanel.llegadaPieza("camionVacio");
247             retrasoGrafica(delay);
248             simPanel.moverRecurso("camion", 260, 550);
249             retrasoGrafica(delay);
250             //simPanel.moverRecurso("camion", 970, 550);
251             mercado.add(paquete); //añadimos a mercado
252             simPanel.llegadaPieza("mercadoLleno");
253             retrasoGrafica(delay);
254             mercado.remove(paquete); //aliminamos de mercado
255             //simPanel.llegadaPieza("mercadoVacio");
256             salidaCajasC++; //aumentamos contador de salidas
257             contMercado++; //aumentamos el contador de salidas
258         }
259     }

```

En la parte de procesos iniciamos de atrás hacia adelante así que iniciamos en almacén donde preguntamos si no está vacío luego recorremos la locación para obtener la entidad actual, luego hacemos todo el proceso de movida de la entidad. Que va desde eliminar de almacén, añadir al camión, mover el camión, eliminar del camión y añadir al mercado que es donde la entidad saldrá de forma definitiva y luego actualizamos los datos de la gráfica.

```

// Almacen
if (!almacen.isEmpty()) {
    for (int al = 0; al < almacen.size(); al++) { //almacen no esta vacio
        Entidad caja = almacen.get(al); //recorremos cada elemento de almacen
        if (!caja.getInArea4()) { //guardamos la entida
            if (almacen.size() >= 6) { //se proceso el elemento
                caja.setInArea4(true); //hay suficientes paquetes para carga
                // Marcar como procesado
                for (int ca = 0; ca < 6 && !almacen.isEmpty(); ca++) { //procesamos 6 paquetes
                    Entidad cajaProcesada = almacen.remove(0); //Eliminar la primera botella
                    textAreaA.append("Se empacó caja " + (ca + 1) + "\n");
                }
                almacen.add(caja);
                procesarOperacion(caja, tiempoActual, "almacen", 0.08, "caja de cerveza");//procesamos carga en almceen
            } else {
                //textAreaA.append("Esperando más cajas con para empaquetar\n");
                break; // Salir del bucle si no se cumplen las condiciones
            }
        }
    }
}

```

Luego en el mismo almacén es donde se combina los paquetes para realizar una entrega es decir se combinan 6 paquetes para realizar una entrega al mercado y de igual forma actualizamos la gráfica después del proceso.

Ese es el proceso general de las locaciones del sistema donde primero se procesan y luego se actualizan los datos a continuación solo pondremos imágenes de los procesos para no ser repetitivos.

```

//Empacado salida
if(!empacado.isEmpty()){
    for(int emp = 0; emp < empacado.size(); emp++){
        Entidad caja = empacado.get(emp);
        if(caja.getTiempo13() == tiempoActual){
            if(almacen.size() < 6){
                empacado.remove(caja);
                simPanel.llegadaPieza("empacadoVacio");
                retrasoGrafica(delay);
                trabajadorCajas.add(caja); //añadimos a operador
                simPanel.llegadaPieza("operador4Lleno");
                retrasoGrafica(delay);
                simPanel.moverRecurso("operadorCajas", 110, 390);
                textAreaA.append("Caja con cerveza salio de empacado\n");
                trabajadorCajas.remove(caja); //eliminamos de operador
                simPanel.moverRecurso("operadorCajas", 110, 540);
                retrasoGrafica(delay);
                simPanel.llegadaPieza("operador4Vacio");
                almacen.add(caja); //añadimos a almacen
                simPanel.llegadaPieza("almacenLleno");
                simPanel.moverRecurso("operadorCajas", 110, 390);
                contAlmacen++; //incrementa el contador de paquetes
            }else{
                //textAreaA.append("Caja con cerveza bloqueada en empacado\n");
                caja.setTiempo13(dosDecimales(tiempoActual+0.01)); //actualizamos el tiempo
            }
        }
    }
}

simPanel.actualizarEstado(

//Empacado
if (!empacado.isEmpty()) { //empacado no esta vacio
    textAreaA.append("Botellas en empacado "+empacado.size()+"\n");
    for (int emp = 0; emp < empacado.size(); emp++) { //recorremos cada entidad
        Entidad botella = empacado.get(emp); //guardamos la entidad
        if (!botella.getInArea3()) { //vemos si se mesclo la entidad
            if (almacenCajas.size() >= 1 && empacado.size() >= botellaPaquete) { //ver si hay cajas y botellas disponibles
                botella.setInArea3(true); // Marcar como procesado
                for (int bot = 0; bot < botellaPaquete; bot++) { //procesamos n botellas
                    Entidad botellaProcesada = empacado.remove(0); //Eliminar la primera botella
                    salidaBotella++; //aumentamos contador de salidas
                    textAreaA.append("Se empacó botella " + (bot + 1) + "\n");
                }
                Entidad caja = almacenCajas.remove(0); //Eliminar la primera caja
                salidaCaja++; //aumentamos contador de salidas
                textAreaA.append("Se utilizó una caja\n");
                empacado.add(botella);
                //textAreaA.append("Cajas producidas: " + contEmpacado + "\n");
                botella.cambiarNombre("Caja de cerveza"); //Cambiamos el nombre a caja de cerveza
                procesarOperacion(botella, tiempoActual, "empacado", 0.17, "caja de cerveza"); //procesamos en empacado
            } else {
                textAreaA.append("Esperando botellas\n");
                break; // Salir del bucle si no se cumplen las condiciones
            }
        }
    }
}

simPanel.actualizarEstado(

//Etiquetado
if(!etiquetado.isEmpty()){
    for(int eti = 0; eti < etiquetado.size(); eti++){
        Entidad botella = etiquetado.get(eti);
        if(botella.getTiempo12() == tiempoActual){
            if(empacado.size() < 6){
                etiquetado.remove(botella);
                simPanel.llegadaPieza("etiquetadoVacio");
                retrasoGrafica(delay);
                textAreaA.append("Botella salio de etiquetado\n");
                empacado.add(botella); //añadimos a empacado
                simPanel.llegadaPieza("empacadoLleno");
                contEmpacado++; //Aumentamos el contador de botellas
            }else{
                //textAreaA.append("Botella bloqueado en etiquetado\n");
                botella.setTiempo12(dosDecimales(tiempoActual+0.01)); //actualizamos el tiempo
            }
        }
    }
}

simPanel.actualizarEstado(

```

```

//Embottellado
if(!embottellado.isEmpty()){ //embottellado no esta vacio
    for(int emb = 0; emb < embottellado.size(); emb++){ //recorremos cada elemento
        Entidad botella = embottellado.get(emb); //guardamos la entidad
        if(botella.getTiempo1() == tiempoActual){ //preguntamos si es tiempo de salida
            if(etiquetado.size() < 6){ //hay espacio en etiquetado
                embottellado.remove(botella); //eliminamos de embottellado
                simPanel.llegadaPieza("embottelladoVacio");
                retrasoGrafica(delay);
                textAreaA.append("Botella salio de embottellado\n");
                etiquetado.add(botella); //añadimos a etiquetado
                simPanel.llegadaPieza("etiquetadoLleno");
                contEtiquetado++; //aumentamos el contador de etiquetado
                procesarOperacion(botella, tiempoActual, "etiquetado", 0.02, "botella");//procesamos en etiquetado
            }else{
                //textAreaA.append("Botella bloqueado en embottellado\n");
                botella.setTiempo1(dosDecimales(tiempoActual+0.01));
            }
        }
    }
}

simPanel.actualizarEstado(

//Inspeccion
if(!inspeccion.isEmpty()){ //inspeccion no esta vacio
    for(int ins = 0; ins < inspeccion.size(); ins++){ //recorremos cada entidad
        Entidad alcohol = inspeccion.get(ins); //guardamos la entidad
        if(alcohol.getTiempo0() == tiempoActual){ //preguntamos si es tiempo de salida
            double random = Math.random(); //generamos un random de 0-1
            if(random > 0.9){ //si esta en el 10% se acepta
                inspeccion.remove(alcohol); //eliminamos alcohol
                simPanel.llegadaPieza("inspeccionVacio");
                retrasoGrafica(delay);
                contRechazado++; //aumentamos el contador de rechazados
                textAreaA.append("Alcohol se rechazado\n");
            }else{
                if(embottellado.size() < 6){ //hay espacio en embottellado
                    inspeccion.remove(alcohol); //eliminamos alcohol de inspeccion
                    simPanel.llegadaPieza("inspeccionVacio");
                    retrasoGrafica(delay);
                    textAreaA.append("Alcohol aceptado y salio de inspeccion\n");
                    embottellado.add(alcohol); //añadimos a embottellado
                    simPanel.llegadaPieza("embottelladoLleno");
                    contEmbottellado++; //aumentamos el contador de embottellado
                    textAreaA.append("botellas en inspeccion"+contEmbottellado+"\n");
                    procesarOperacion(alcohol, tiempoActual, "embottellado", 0.1, "alcohol");//procesamos en embottellado
                }else{
                    //textAreaA.append("Alcohol bloqueado en inspeccion\n");
                    alcohol.setTiempo0(dosDecimales(tiempoActual+0.01)); //Actualizamos el tiempo
                }
            }
        }
    }
}

```

En esta parte en inspección es donde decidimos si la muestra de alcohol se acepta o se rechaza utilizando un número randomico que va del 0 al 1 para comprobar el 90% y el 10% si se rechaza se elimina la entidad, pero si se acepta se manda la entidad a procesarse en la siguiente área.

```

//Madurado
if(!maduracion.isEmpty()){ //maduracion no esta vacio
    for(int ma = 0; ma < maduracion.size(); ma++){ //recorremos cada entidad
        Entidad alcohol = maduracion.get(ma); //guardamos a entidad
        if(alcohol.getTiempo9() == tiempoActual){ //preguntamos si es tiempo de salida
            if(inspeccion.size() < 3){ //hay espacio en inspeccion
                maduracion.remove(alcohol); //eliminamos de maduracion
                simPanel.llegadaPieza("maduracionVacio");
                retrasoGrafica(delay);
                textAreaA.append("Alcohol salio de maduración\n");
                inspeccion.add(alcohol); //añadimos a inspeccion
                simPanel.llegadaPieza("inspeccionLleno");
                contIns++; //aumentamos el contador de inspeccion
                procesarOperacion(alcohol, tiempoActual, "inspeccion", 0.5, "alcohol");//procesamos en inspeccion
            }else{
                //textAreaA.append("Alcohol bloqueado en maduración\n");
                alcohol.setTiempo9(dosDecimales(tiempoActual+0.01));
            }
        }
    }
}

```



```

//Fermetado salida
if(!fermentado.isEmpty()){
    for(int fer = 0; fer < fermentado.size(); fer++){
        Entidad alcohol = fermentado.get(fer);
        if(alcohol.getTiempo8() == tiempoActual){
            if(maduracion.size() < 10){
                fermentado.remove(alcohol);
                simPanel.llegadaPieza("fermentadoVacio");
                retrasoGrafica(delay);
                textAreaA.append("Alcohol salio de fermentado\n");
                maduracion.add(alcohol);
                simPanel.llegadaPieza("maduracionLleno");
                contMaduracion++;
                procesarOperacion(alcohol, tiempoActual, "madurado", 1.5, "alcohol");//procesamos en maduracion
            }else{
                //textAreaA.append("Alcohol bloqueado en fermentado\n");
                alcohol.setTiempo8(dosDecimales(tiempoActual+0.01));
            }
        }
    }
}
simPanel.actualizarEstado(

```

Luego tenemos otra área importante que es el fermentado donde esta parte se encarga de las salidas de alcohol que ya se mezclaron con los ingredientes y luego se manda a actualizar los datos.

```

506 if (!fermentado.isEmpty()) {
507     for (int fer = 0; fer < fermentado.size(); fer++){
508         Entidad mosto = fermentado.get(fer);
509         if (!mosto.getInArea2()) {
510             if (siloLevadura.size() >= 2 && fermentado.size() >= litrosMosto){//si hay levadura y mosto suficiente
511                 mosto.setInArea2(true);
512                 for (int mos = 0; mos < litrosMosto && !fermentado.isEmpty(); mos++){//procesamos litros mosto
513                     Entidad mostoProcesado = fermentado.remove(0); //Eliminar el primer mosto
514                     salidaMosto++;
515                     textAreaA.append("Se procesó mosto " + (mos + 1) + "\n");
516                 }
517                 for (int lv = 0; lv < 2 && !siloLevadura.isEmpty(); lv++){//procesamos levadura necesaria
518                     retrasoGrafica(delay);
519                     Entidad levadura = siloLevadura.remove(0); //Eliminar la primera levadura
520                     salidaLevadura++;
521                     trabajadorLevadura.add(levadura);
522                     simPanel.llegadaPieza("operador3Lleno");
523                     retrasoGrafica(delay);
524                     simPanel.moverRecurso("operadorLevadura", 1000, 390);
525                     trabajadorLevadura.remove(levadura); //eliminamos de operador
526                     simPanel.llegadaPieza("operador3Vacio");
527                     retrasoGrafica(delay);
528                     simPanel.moverRecurso("operadorLevadura", 1160, 210);
529                     textAreaA.append("Se procesó levadura " + (lv + 1) + "\n");
530                 }
531                 textAreaA.append("Alcohol producido: " + contAlcohol + "\n");
532                 mosto.cambiarNombre("Alcohol");
533                 fermentado.add(mosto);
534                 procesarOperacion(mosto, tiempoActual, "fermentado", 2, "mosto");//procesamos en fermentado
535             } else {
536                 //textAreaA.append("Esperando más mosto o levadura\n");
537                 break; // Salir del bucle si no se cumplen las condiciones
538             }
539         }
540     }

```

En la parte de fermentado que es donde se cumplen las condiciones de las mezclas como las cantidades si es así recién realizamos los procesos correspondientes para poder realizarlo y como siempre al final se manda el mesclado a la lista de nuevo para que sea analizada y salga en la hora correspondiente.

Básicamente todos los procesos funcionan de esa manera por lo que iremos directamente al final de la simulación a la parte de resultados.

```

850 double area6V = dosDecimales(((double)contArea6V / (duracion * 100)) * 100);
851 double area7L = dosDecimales(((double)contArea7L / (duracion * 100)) * 100);
852 double area7V = dosDecimales(((double)contArea7V / (duracion * 100)) * 100);
853 double area8L = dosDecimales(((double)contArea8L / (duracion * 100)) * 100);
854 double area8V = dosDecimales(((double)contArea8V / (duracion * 100)) * 100);
855 double area9L = dosDecimales(((double)contArea9L / (duracion * 100)) * 100);
856 double area9V = dosDecimales(((double)contArea9V / (duracion * 100)) * 100);
857 double area10L = dosDecimales(((double)contArea10L / (duracion * 100)) * 100);
858 double area10V = dosDecimales(((double)contArea10V / (duracion * 100)) * 100);
859 double area11L = dosDecimales(((double)contArea11L / (duracion * 100)) * 100);
860 double area11V = dosDecimales(((double)contArea11V / (duracion * 100)) * 100);
861 double area12L = dosDecimales(((double)contArea12L / (duracion * 100)) * 100);
862 double area12V = dosDecimales(((double)contArea12V / (duracion * 100)) * 100);
863 double area13L = dosDecimales(((double)contArea13L / (duracion * 100)) * 100);
864 double area13V = dosDecimales(((double)contArea13V / (duracion * 100)) * 100);
865 double area14L = dosDecimales(((double)contArea14L / (duracion * 100)) * 100);
866 double area14V = dosDecimales(((double)contArea14V / (duracion * 100)) * 100);
867 double area15L = dosDecimales(((double)contArea15L / (duracion * 100)) * 100);
868 double area15V = dosDecimales(((double)contArea15V / (duracion * 100)) * 100);
869 double area16L = dosDecimales(((double)contArea16L / (duracion * 100)) * 100);
870 double area16V = dosDecimales(((double)contArea16V / (duracion * 100)) * 100);
871 double area17L = dosDecimales(((double)contArea17L / (duracion * 100)) * 100);
872 double area17V = dosDecimales(((double)contArea17V / (duracion * 100)) * 100);
873 double area18L = dosDecimales(((double)contArea18L / (duracion * 100)) * 100);
874 double area18V = dosDecimales(((double)contArea18V / (duracion * 100)) * 100);
875 double area19L = dosDecimales(((double)contArea19L / (duracion * 100)) * 100);
876 double area19V = dosDecimales(((double)contArea19V / (duracion * 100)) * 100);
877 BarChartExample example = new BarChartExample(
878     area1L, area1V, area2L, area2V, area3L, area3V, area4L, area4V, area5L, area5V,
879     area6L, area6V, area7L, area7V, area8L, area8V, area9L, area9V, area10L, area10V,
880     area11L, area11V, area12L, area12V, area13L, area13V, area14L, area14V, area15L, area15V,
881     area16L, area16V, area17L, area17V, area18L, area18V, area19L, area19V
882 );
883 example.setVisible(true);
884

```

Para la parte de porcentajes mandamos a imprimir una grafica de barras con los datos recolectados a través del periodo de simulación.

```

885 //Tabla
886 Object[][] data = {
887     {"Arribos", ""},
888     {"Arribo de cebada", contSiloCebada},
889     {"Arribo fallido de cebada", contSiloCebadaF},
890     {"Arribo de lupulo", contSiloLupulo},
891     {"Arribo fallido de lupulo", contSiloLupuloF},
892     {"Arribo de levadura", contSiloLevadura},
893     {"Arribo fallido de levadura", contSiloLevaduraF},
894     {"Arribo de caja vacia", contAlmacenCaja},
895     {"Arribo fallido de caja vacia", contAlmacenCajaF},
896
897     // Cantidad procesada en locaciones
898     {"Cantidades procesadas por cada locacion", ""},
899     {"En malteado", contMalteado},
900     {"En secado", contSecado},
901     {"En molienda", contMolienda},
902     {"En maserado", contMaserado},
903     {"En filtrado", contFiltrado},
904     {"En cocción", contMosto},
905     {"En enfriado", contEnfriado},
906     {"En fermentado", contAlcohol},
907     {"En maduración", contMaduracion},
908     {"En inspección", contIns},
909     {"Rechazados", contRechazado},
910     {"En embotellado", contEmbotellado},
911     {"En etiquetado", contEtiquetado},
912     {"En empacado", contEmpacado},
913     {"En almacen", contAlmacen},
914     {"En mercado", contMercado},

```

Por la parte de datos generales también manda a imprimir los datos relevantes para el ejercicio en una misma tabla.


```

969
970 class SimulacionPanel extends JPanel {
971     //Locaciones
972     private Area siloGrano = new Area("Silo de grano", 10, 20, 100, 100, Color.GREEN);
973     private Area malteado = new Area("Malteado", 150, 20, 100, 100, Color.GREEN);
974     private Area secado = new Area("Secado", 300, 20, 100, 100, Color.GREEN);
975     private Area molienda = new Area("Molienda", 450, 20, 100, 100, Color.GREEN);
976     private Area maserado = new Area("Maserado", 600, 20, 100, 100, Color.GREEN);
977     private Area filtrado = new Area("Filtrado", 750, 20, 100, 100, Color.GREEN);
978     private Area siloLupulo = new Area("Silo de Lupulo", 600, 200, 100, 100, Color.GREEN);
979     private Area coccion = new Area("Coccion", 900, 20, 100, 100, Color.GREEN);
980     private Area enfriado = new Area("Enfriado", 900, 200, 100, 100, Color.GREEN);
981     private Area siloLevadura = new Area("Silo de levadura", 1050, 200, 100, 100, Color.GREEN);
982     private Area fermentado = new Area("fermentado", 900, 380, 100, 100, Color.GREEN);
983     private Area madurado = new Area("Madurado", 750, 380, 100, 100, Color.GREEN);
984     private Area inspeccion = new Area("Inspeccion", 600, 380, 100, 100, Color.GREEN);
985     private Area embotellado = new Area("Embotellado", 450, 380, 100, 100, Color.GREEN);
986     private Area etiquetado = new Area("Etiquetado", 300, 380, 100, 100, Color.GREEN);
987     private Area almacenCajas = new Area("Almacen de cajas", 150, 200, 100, 100, Color.GREEN);
988     private Area empacado = new Area("Empacado", 150, 380, 100, 100, Color.GREEN);
989     private Area almacen = new Area("Almacen", 150, 530, 100, 100, Color.GREEN);
990     private Area mercado = new Area("Mercado", 1050, 530, 100, 100, Color.GREEN);
991     //Recursos
992     private Area operadorRecepcion = new Area("", 550, 210, 40, 80, Color.GREEN);
993     private Area operadorLupulo = new Area("", 710, 210, 40, 80, Color.GREEN);
994     private Area operadorLevadura = new Area("", 1160, 210, 40, 80, Color.GREEN);
995     private Area operadorCaja = new Area("", 110, 390, 40, 80, Color.GREEN);
996     private Area camion = new Area("", 260, 550, 80, 40, Color.GREEN);
997     //Contadores
998     private Area contSiloGrano = new Area("", 10, 130, 100, 20, Color.GRAY);
999     private Area contMalteado = new Area("", 150, 130, 100, 20, Color.GRAY);
1000     private Area contSecado = new Area("", 300, 130, 100, 20, Color.GRAY);
1001     private Area contMolienda = new Area("", 450, 130, 100, 20, Color.GRAY);
1002     private Area contMaserado = new Area("", 600, 130, 100, 20, Color.GRAY);

```

Luego tenemos la parte grafica donde construimos los componentes que se visualizaran en la gráfica de simulación.

```

1041 //variables de imagenes
1042 private BufferedImage siloImg, malteadoImg, secadoImg, moliendaImg, maseradoImg, filtradoImg,
1043     coccionImg, enfriadoImg, fermentadoImg, maduradoImg, inspeccionImg, embotelladoImg, etiquetadoImg,
1044     almacenCajasImg, empacadoImg, almacenImg, mercadoImg, operadorImg, camionImg, granoImg, lupuloImg,
1045     levaduraImg, cajaImg, mostoImg, alcoholImg, botellaImg, cajaCervezaImg, paqueteImg,
1046     operaRecepcionImg, operaLupuloImg, operaLevaduraImg, operaCajaImg, camionCargaImg;
1047
1048 public SimulacionPanel() {
1049     this.setPreferredSize(new Dimension(800, 600));
1050     this.setBackground(Color.WHITE);
1051     // Cargar la imagen al inicializar el panel
1052     try {
1053         siloImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/silo.jpeg"));
1054         malteadoImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/malteado.jpeg"));
1055         secadoImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/secadol.jpg"));
1056         moliendaImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/molienda.jpeg"));
1057         maseradoImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/maserado.jpeg"));
1058         filtradoImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/filtrado.jpeg"));
1059         coccionImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/coccion.jpeg"));
1060         enfriadoImg = ImageIO.read(getClass().getResourceAsStream("/com/mycompany/proyecto3/img/enfriado.png"));

```

Luego tenemos las variables donde se guardarán las imágenes que se utilizan para la simulación.

```

1090
1091 //Damos imagen de fondo a los sitios fijos
1092 siloGrano.setBackgroundImage(siloImg);
1093 malteado.setBackgroundImage(malteadoImg);
1094 secado.setBackgroundImage(secadoImg);
1095 molienda.setBackgroundImage(moliendaImg);
1096 maserado.setBackgroundImage(maseradoImg);
1097 filtrado.setBackgroundImage(filtradoImg);
1098 coccion.setBackgroundImage(coccionImg);
1099 siloLupulo.setBackgroundImage(siloImg);
1100 enfriado.setBackgroundImage(enfriadoImg);
1101 siloLevadura.setBackgroundImage(siloImg);
1102 fermentado.setBackgroundImage(fermentadoImg);
1103 madurado.setBackgroundImage(maduradoImg);
1104 inspeccion.setBackgroundImage(inspeccionImg);
1105 embotellado.setBackgroundImage(embotelladoImg);
1106 etiquetado.setBackgroundImage(etiquetadoImg);
1107 almacenCajas.setBackgroundImage(almacenCajasImg);
1108 empacado.setBackgroundImage(empacadoImg);
1109 almacen.setBackgroundImage(almacenImg);
1110 mercado.setBackgroundImage(mercadoImg);

```

Luego se asignan las imágenes a las áreas respectivas.

```

119
120 public void llegadaPieza(String nombre) {
121     // Cambia la imagen de fondo del área seleccionada y configura el temporizador para volver al color original
122     switch (nombre) {
123         case "siloGrano":
124             CsiloGrano.setBackgroundImage(granoImg);
125             repaint();
126             new Timer(1000, e -> {
127                 CsiloGrano.setBackgroundImage(null); // Quita la imagen de fondo
128                 repaint();
129             }).start();
130             break;
131         case "siloLupulo":
132             CsiloLupulo.setBackgroundImage(lupuloImg);
133             repaint();
134             new Timer(1000, e -> {
135                 CsiloLupulo.setBackgroundImage(null); // Quita la imagen de fondo
136                 repaint();
137             }).start();
138             break;

```

Luego tenemos el método que se encarga de cambiar las imágenes según sea necesario para la simulación.

```

1320 public void moverRecurso(String nombre, int nuevaX, int nuevaY) {
1321     switch (nombre) {
1322         case "operadorRecepcion":
1323             operadorRecepcion.setX(nuevaX);
1324             operadorRecepcion.setY(nuevaY);
1325             repaint(); // Actualiza la gráfica
1326             break;
1327         case "operadorLupulo":
1328             operadorLupulo.setX(nuevaX);
1329             operadorLupulo.setY(nuevaY);
1330             repaint(); // Actualiza la gráfica
1331             break;
1332         case "operadorLevadura":
1333             operadorLevadura.setX(nuevaX);
1334             operadorLevadura.setY(nuevaY);
1335             repaint(); // Actualiza la gráfica
1336             break;
1337         case "operadorCajas":
1338             operadorCaja.setX(nuevaX);
1339             operadorCaja.setY(nuevaY);
1340             repaint(); // Actualiza la gráfica
1341             break;
1342         case "camion":
1343             camion.setX(nuevaX);
1344             camion.setY(nuevaY);
1345             repaint(); // Actualiza la gráfica
1346             break;
1347         default:
1348             return;

```

Luego tenemos el método para mover los componentes en este caso lo operadores y el camión de entrega de paquetes de cerveza.

```

1352 public void actualizarEstado(
1353     int contCebada, int contLupulo, int contLevadura, int contCaja,
1354     int contMalteado, int contSecado, int contMolienda, int contMaserado, int contFiltrado,
1355     int contMosto, int contEnfriado, int contAlcohol, int contMaduracion, int contIns,
1356     int contRechazado, int contEmbotellado, int contEtiquetado, int contEmpacado,
1357     int contAlmacen, int contMercado) {
1358
1359     // Actualiza los contadores de cada área
1360     this.contSiloGrano.setContador(contCebada);
1361     this.contSiloLupulo.setContador(contLupulo);
1362     this.contSiloLevadura.setContador(contLevadura);
1363     this.contAlmacenCajas.setContador(contCaja);
1364     this.contRechazados.setContador(contRechazado);
1365     this.contMalteado.setContador(contMalteado);
1366     this.contSecado.setContador(contSecado);
1367     this.contMolienda.setContador(contMolienda);
1368     this.contMaserado.setContador(contMaserado);
1369     this.contFiltrado.setContador(contFiltrado);

```

Método para actualizar los datos visibles de la gráfica como las cantidades.

```

471
472 public double arribo (List<Entidad> locacion, int capacidad, String recurso, double intervalo, double tiempoActual){
473     double res = 0;
474     if(locacion.size() < capacidad){
475         if(null != recurso)switch (recurso) {
476             case "cebada" -> contSiloCebada++;
477             case "lupulo" -> contSiloLupulo++;
478             case "levadura" -> contSiloLevadura++;
479             case "caja vacia" -> contAlmacenCaja++;
480             default -> {
481                 }
482             }
483         textAreaA.append("Llego 1 kg "+recurso+"\n"); //imprime hora
484         Entidad entidad = new Entidad(recurso,1, false, false, false); //con sus datos
485         locacion.add(entidad); //y los ponemos en estraje
486         res = dosDecimales(tiempoActual + intervalo);
487     }else{
488         if(null != recurso)switch (recurso) {
489             case "cebada" -> contSiloCebadaF++;
490             case "lupulo" -> contSiloLupuloF++;
491             case "levadura" -> contSiloLevaduraF++;
492             case "caja vacia" -> contAlmacenCajaF++;
493             default -> {
494                 }
495             }
496         textAreaA.append("Arribo fallido 1 "+recurso+"\n");
497         res = dosDecimales(tiempoActual + intervalo); //mandamos el tiempo del siguiente pero vacio
498     }
499     return res;
500 }

```

Luego tenemos el método de arribo donde según a la entidad que esta entrando este realiza la acción de añadirlo al lugar donde corresponde y también aumenta el numero de contadores en el arribo.

```

1501
1502 public void procesarOperacion(Entidad grano, double tiempoActual, String operacion, double proceso, String nombre) {
1503     //textAreaA.append("El "+nombre+" tarda una "+proceso+" horas en " + operacion + "\n");
1504     double tiempoSalida = dosDecimales(tiempoActual + proceso); // procesamos en tiempo de horas
1505
1506     // Actualizar el tiempo correspondiente según la operación
1507     switch (operacion.toLowerCase()) {
1508         case "malteado":
1509             grano.setTiempo1(tiempoSalida);
1510             break;
1511         case "secado":
1512             grano.setTiempo2(tiempoSalida);
1513             break;
1514         case "molienda":

```

Y finalmente tenemos el método de proceso que es el que asigna los tiempos correspondientes a las diferentes operaciones de las locuciones a las entidades correspondientes.

Resultados:

Pro Model:

a) Analizar cuantas entidades fueron procesadas o salieron de cada locación:

| Nombre | Total Cambios | Tiempo Por cambio Promedio (Min) | Valor Mínimo | Valor Máximo | Valor Actual | Valor Promedio |
|------------------|---------------|----------------------------------|--------------|--------------|--------------|----------------|
| contSiloCebada | 124,00 | 33,87 | 0,00 | 124,00 | 124,00 | 68,02 |
| contMalteado | 121,00 | 34,56 | 0,00 | 121,00 | 121,00 | 65,68 |
| contSecado | 118,00 | 35,44 | 0,00 | 118,00 | 118,00 | 62,72 |
| contMolieda | 115,00 | 36,36 | 0,00 | 115,00 | 115,00 | 59,81 |
| contMaserado | 113,00 | 36,99 | 0,00 | 113,00 | 113,00 | 57,89 |
| contFiltrado | 110,00 | 38,00 | 0,00 | 110,00 | 110,00 | 55,05 |
| contCoccion | 108,00 | 38,71 | 0,00 | 108,00 | 108,00 | 53,51 |
| contSiloLupulo | 400,00 | 10,50 | 0,00 | 400,00 | 400,00 | 190,50 |
| contEnfriado | 98,00 | 42,66 | 0,00 | 98,00 | 98,00 | 45,67 |
| contSiloLevadura | 28,00 | 141,43 | 0,00 | 28,00 | 28,00 | 17,71 |
| contFermentado | 96,00 | 43,34 | 0,00 | 96,00 | 96,00 | 44,02 |
| contMaduracion | 9,00 | 448,99 | 0,00 | 9,00 | 9,00 | 3,77 |
| contInspeccion | 9,00 | 458,99 | 0,00 | 9,00 | 9,00 | 3,58 |
| contRechasada | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| contEnbotellado | 9,00 | 462,32 | 0,00 | 9,00 | 9,00 | 3,51 |
| contEtiquetado | 9,00 | 462,99 | 0,00 | 9,00 | 9,00 | 3,50 |
| contEnpacado | 9,00 | 463,10 | 0,00 | 9,00 | 9,00 | 3,50 |
| contAlmacenCajas | 31,00 | 95,81 | 0,00 | 31,00 | 31,00 | 27,19 |
| contAlmcen | 1,00 | 2.978,12 | 0,00 | 1,00 | 1,00 | 0,29 |
| contMercado | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |

Como se puede ver en la columna cantidad actual tenemos que en los silos de cebada se procesaron 124 kilogramos de grano, en el de malteado se procesaron 121, en secado 118, en molienda 115, en macerado 113, en filtrado 110, en cocción 108 litros, en silo de lúpulo se procesaron 400 kilogramos de lúpulo, en enfriado se procesaron 98 litros, en el silo de levadura se procesaron 28 kilogramos, en fermentado 96 litros, en maduración 9 litros, en inspección 9 litros, no existe rechazados, en embotellado, etiquetado, empacado se procesaron 9 botellas, en el almacén de cajas se procesaron 31 cajas en almacén solo 1 y en mercado 0.

b) Cuantas se encuentran en proceso:

| Nombre | Capacidad | Total Entradas | Contenido Promedio | Contenido Máximo | Contenido Actual | % Utilización |
|------------------|------------|----------------|--------------------|------------------|------------------|---------------|
| Silo cebada | 3,00 | 124,00 | 2,34 | 3,00 | 3,00 | 77,94 |
| Malteado | 3,00 | 121,00 | 2,96 | 3,00 | 3,00 | 98,54 |
| Secado | 3,00 | 118,00 | 2,90 | 3,00 | 3,00 | 96,81 |
| Molienda | 2,00 | 115,00 | 1,92 | 2,00 | 2,00 | 95,86 |
| Maserado | 3,00 | 113,00 | 2,84 | 3,00 | 3,00 | 94,58 |
| Filtrado | 2,00 | 110,00 | 1,54 | 2,00 | 2,00 | 77,23 |
| Coccion | 10,00 | 108,00 | 7,84 | 10,00 | 10,00 | 78,35 |
| Silo lupulo | 10,00 | 400,00 | 0,83 | 10,00 | 0,00 | 8,31 |
| Enfriado | 10,00 | 98,00 | 1,65 | 4,00 | 2,00 | 16,52 |
| Silo levadura | 10,00 | 28,00 | 9,66 | 10,00 | 10,00 | 96,59 |
| Fermentado | 10,00 | 96,00 | 6,33 | 10,00 | 6,00 | 63,25 |
| Maduracion | 10,00 | 9,00 | 0,19 | 1,00 | 0,00 | 1,93 |
| Inspeccion | 3,00 | 9,00 | 0,06 | 1,00 | 0,00 | 2,14 |
| Enbotellado | 6,00 | 9,00 | 0,01 | 1,00 | 0,00 | 0,21 |
| Etiquetado | 6,00 | 9,00 | 0,00 | 1,00 | 0,00 | 0,04 |
| Almacen de cajas | 30,00 | 31,00 | 26,89 | 30,00 | 30,00 | 89,64 |
| Empacado | 6,00 | 9,00 | 1,75 | 6,00 | 3,00 | 29,19 |
| Almacenaje | 6,00 | 1,00 | 0,29 | 1,00 | 1,00 | 4,85 |
| Mercado | 999.999,00 | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |

Vemos que actualmente en proceso hay 3 kilogramos en silo de cebada, 3 en malteado, 3 en secado, 2 en molienda, 3 en macerado, 2 en filtrado, 10 litros en cocción, 2 litros en enfriado, en silo de levadura hay 10 kilogramos, en fermentado hay 6 litros, en almacén de cajas hay 30 cajas en empacado hay 3 botellas y en almacenaje hay 1 caja con cerveza.

En total en proceso hay 16 kilogramos de cebada procesándose actualmente, 18 litros de mosto procesando actualmente, hay 10 kilogramos de levadura procesándose actualmente, hay 30 cajas, 3 botellas y 1 caja con cerveza.

c) Cuantas salieron del sistema:

| Nombre | Total Salidas | Cantidad actual En Sistema | Tiempo En Sistema Promedio (Min) | Tiempo Esperando Promedio (Min) | Tiempo de Bloqueo Promedio (Min) |
|--------------------|---------------|----------------------------|----------------------------------|---------------------------------|----------------------------------|
| Grano cebada | 0,00 | 26,00 | 0,00 | 0,00 | 0,00 |
| Lupulo | 399,00 | 1,00 | 9,03 | 8,69 | 0,00 |
| Mosto | 90,00 | 8,00 | 1.043,34 | 414,48 | 207,19 |
| Levadura | 18,00 | 10,00 | 1.670,71 | 1.670,28 | 0,00 |
| Alcohol | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| Botella de cerveza | 6,00 | 3,00 | 1.247,64 | 1.000,64 | 0,00 |
| Caja vacia | 1,00 | 30,00 | 2.967,92 | 2.967,92 | 0,00 |
| Caja con cerveza | 0,00 | 1,00 | 0,00 | 0,00 | 0,00 |

Salieron 399 de lúpulo, 90 de mosto, 18 de levadura, 6 botellas de cerveza y 1 caja vacía.

d) **Cuál es el tiempo promedio en el sistema:**

| Nombre | Total Salidas | Cantidad actual En Sistema | Tiempo En Sistema Promedio (Min) | Tiempo Esperando Promedio (Min) | Tiempo de Bloqueo Promedio (Min) |
|--------------------|---------------|----------------------------|----------------------------------|---------------------------------|----------------------------------|
| Grano cebada | 0,00 | 26,00 | 0,00 | 0,00 | 0,00 |
| Lupulo | 399,00 | 1,00 | 9,03 | 8,69 | 0,00 |
| Mosto | 90,00 | 8,00 | 1.043,34 | 414,48 | 207,19 |
| Levadura | 18,00 | 10,00 | 1.670,71 | 1.670,28 | 0,00 |
| Alcohol | 0,00 | 0,00 | 0,00 | 0,00 | 0,00 |
| Botella de cerveza | 6,00 | 3,00 | 1.247,64 | 1.000,64 | 0,00 |
| Caja vacia | 1,00 | 30,00 | 2.967,92 | 2.967,92 | 0,00 |
| Caja con cerveza | 0,00 | 1,00 | 0,00 | 0,00 | 0,00 |

Vemos que el lupulo tiene un promedio de 9.03, el mosto 1043.34 minutos, la levadura 1670.71 minutos, la botella un promedio de 1247.64 y la caja vacía un promedio de 2967.92 minutos.

Java:

a) **Analizar cuantas entidades fueron procesadas o salieron de cada locación:**

| Arribos | |
|---|-----|
| Arribo de cebada | 122 |
| Arribo fallido de cebada | 45 |
| Arribo de lupulo | 392 |
| Arribo fallido de lupulo | 20 |
| Arribo de levadura | 28 |
| Arribo fallido de levadura | 185 |
| Arribo de caja vacia | 31 |
| Arribo fallido de caja vacia | 109 |
| Cantidades procesadas por cada locacion | |
| En malteado | 119 |
| En secado | 116 |
| En molienda | 113 |
| En maserado | 111 |
| En filtrado | 108 |
| En cocción | 98 |
| En enfriado | 96 |
| En fermentado | 95 |
| En maduración | 9 |
| En inspección | 9 |
| Rechazados | 1 |
| En embotellado | 7 |
| En etiquetado | 7 |
| En empacado | 7 |
| En almacen | 1 |
| En mercado | 0 |

Como se puede ver tenemos que en los silos de cebada se procesaron 122 kilogramos de grano, en el de malteado se procesaron 119, en secado 116, en molienda 113, en macerado 111, en filtrado 108, en cocción 98 litros, en silo de lúpulo se procesaron 392 kilogramos de lúpulo, en enfriado se procesaron 96 litros, en el silo de levadura se procesaron 28 kilogramos, en fermentado 95 litros, en maduración 9 litros, en inspección 9 litros, 1 es rechazado, en embotellado, etiquetado, empacado se procesaron 7 botellas, en el almacén de cajas se procesaron 31 cajas en almacén solo 1 y en mercado 0.

b) Cuantas se encuentran en proceso:

| Entidades en proceso | |
|----------------------|----|
| En silo de cebada | 3 |
| En Malteado | 3 |
| En secado | 3 |
| En molienda | 2 |
| En maserado | 3 |
| En filtrado | 2 |
| En silo de lupulo | 0 |
| En cocción | 10 |
| En enfriado | 1 |
| En silo de levadura | 10 |
| En fermentación | 5 |
| En maduración | 0 |
| En inspección | 1 |
| En embotellado | 0 |
| En etiquetado | 0 |
| En almacen de cajas | 30 |
| En empacado | 1 |
| En almacen | 1 |
| En mercado | 0 |

Vemos que actualmente en proceso hay 3 kilogramos en silo de cebada, 3 en malteado, 3 en secado, 2 en molienda, 3 en macerado, 2 en filtrado, 10 litros en cocción, 1 litros en enfriado, en silo de levadura hay 10 kilogramos, en fermentado hay 5 litros, en inspección hay 1 litro, en almacén de cajas hay 30 cajas en empacado hay 1 botellas y en almacenaje hay 1 caja con cerveza.

En total en proceso hay 16 kilogramos de cebada procesándose actualmente, 18 litros de mosto procesando actualmente, hay 10 kilogramos de levadura procesándose actualmente, hay 30 cajas, 3 botellas y 1 caja con cerveza.

c) Cuantas salieron del sistema:

| Total salidas | |
|--------------------------------|-----|
| Salidas de grano de cebada | 0 |
| Salidas de lupulo | 392 |
| Salidas de Mosto | 90 |
| Salidas de levadura | 18 |
| Salidas de alcohol | 0 |
| Salidas de botellas de cerveza | 6 |
| Salidas de caja vacía | 1 |
| Salidas de cajas de cerveza | 0 |

Como se puede ver salieron 392 de lúpulo, 90 de mosto, 18 de levadura, 6 botellas de cerveza y una caja vacía.

d) **Cuál es el tiempo promedio en el sistema:**

| Tiempo promedio en sistema en minutos | |
|--|--------|
| Tiempo promedio en sistema de grano d... | 0.0 |
| Tiempo promedio en sistema de lupulo | 8.9 |
| Tiempo promedio en sistema de Mosto | 1041.1 |
| Tiempo promedio en sistema de levadura | 1670.0 |
| Tiempo promedio en sistema de alcohol | 0.0 |
| Tiempo promedio en sistema de botella... | 1245.3 |
| Tiempo promedio en sistema de caja va... | 2967.1 |
| Tiempo promedio en sistema de cajas d... | 0.0 |

El tiempo promedio de las entidades en sistema es de 8.9 para el lúpulo, 1041.1 minutos para el mosto, 1670 minutos para la levadura, 1245.3 para las botellas y 2967.1 para las cajas vacías.

Tabla de comparación:

| | Pro model | Java |
|---------------------------------------|--|---|
| Entidades procesadas en cada locación | Silo de cebada: 124 kg Malteado: 121 kg Secado: 118 kg Molienda: 115 kg Macerado: 113 kg Filtrado: 110 kg Cocción: 108 litros Silo de lúpulo: 400 kg Enfriado: 98 litros Silo de levadura: 28 kg Fermentado: 96 litros Maduración: 9 litros Inspección: 9 litros Embotellado: 9 botellas Etiquetado: 9 botellas Empacado: 9 botellas Almacén de cajas: 31 cajas Almacén: 1 caja Mercado: 0 cajas | Silo de cebada: 122 kg Malteado: 119 kg Secado: 116 kg Molienda: 113 kg Macerado: 111 kg Filtrado: 108 kg Cocción: 98 litros Silo de lúpulo: 392 kg Enfriado: 96 litros Silo de levadura: 28 kg Fermentado: 95 litros Maduración: 9 litros Inspección: 9 litros Embotellado: 7 botellas Etiquetado: 7 botellas Empacado: 7 botellas Almacén de cajas: 31 cajas Almacén: 1 caja Mercado: 0 cajas |
| Entidades en proceso | Cebada: 16 kg Mosto: 18 litros Levadura: 10kg Cajas: 30 Botellas: 3 Cajas con cerveza: 1 | Cebada: 16 kg Mosto: 16 litros Levadura: 10kg Cajas: 30 Botellas: 1 Cajas con cerveza: 1 |
| Entidades que salieron | Lupulo: 399 kg Mosto: 90 kg Levadura: 18 kg Botellas: 6 Caja vacía: 1 | Lupulo: 392 kg Mosto: 90 kg Levadura: 18 kg Botellas: 6 Caja vacía: 1 |

| | | |
|------------------------------|--|---|
| Tiempo promedio de entidades | Lúpulo: 9.03 min Mosto: 1043.34 min Levadura: 1670.71 min Botella: 1247.64 min Caja vacía: 2967.92 min | Lúpulo: 8.9 min Mosto: 1041.1 min Levadura: 1670 min Botella: 1245.3 min Caja vacía: 2967.1 min |
|------------------------------|--|---|

Conclusión:

En conclusión, podemos decir que el modelo de producción de cerveza es un proceso amplio y que toma su tiempo por lo que se vio en el modelo las horas de procesamiento del producto antes de ser cerveza son las que mas tiempo les tomo realizar y también viendo que en las combinaciones podemos decir que dependiendo a las combinaciones que hagamos el área de fermentado podemos producir mas cajas de cerveza o menos también podemos ver que el área de cocción influye en la cantidad.

Videos del proyecto:

Pro Model:

<https://youtu.be/iSxkTONbH4w>

Java:

<https://youtu.be/mcnmHU1FkIQ>