

Universidad de San Carlos de Guatemala

Escuela de Ciencias y Sistemas

Facultad de Ingeniería

Introducción a la programación y Computación 1

Segundo Semestre 2024

Catedrático:

Tutor académico:



FIUSAC
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

PRÁCTICA 1

Simulación de Línea de Producción Automatizada

Objetivos

Generales

- Familiarizar al estudiante con el lenguaje de programación Java.
- El estudiante aplique los conocimientos adquiridos en el curso de Introducción a la programación y computación 1.
- Elaborar la lógica para presentar una solución a la problemática planteada.

Específicos

- Utilizar el lenguaje de programación Java como herramienta de desarrollo de software.
- Aplicación de conceptos de programación orientada a objetos.
- Construcción de aplicaciones con interfaz gráfica.
- Implementación de sentencias de control, ciclos, records y librerías de interfaz gráfica.
- Aplicación de conceptos de programación para crear herramientas administrativas y de automatización.
- Implementación de soluciones con el uso de hilos.
- Crear respaldos de información y restablecerla con el uso de serialización.

Descripción General

Automatizaciones IPCTEC, una empresa líder en soluciones de automatización industrial, enfrenta desafíos significativos en la gestión de sus líneas de producción automatizadas. Con la necesidad de mejorar la eficiencia y la coordinación en la producción de diversos productos, es crucial contar con un sistema que simule y gestione de manera precisa el flujo de trabajo a través de diferentes estaciones de producción. Esta simulación debe permitir una visualización clara del progreso de cada etapa, garantizando una integración fluida entre las distintas fases del proceso.

Aplicación

El objetivo principal es desarrollar una simulación de una línea de producción en la que se incluyan estaciones de trabajo como ensamblaje, pintura y empaquetado. Cada una de estas estaciones debe operar como hilos independientes, trabajando de manera sincronizada para asegurar que los productos avancen a través de la línea de producción sin problemas. La interfaz gráfica proporcionará una visualización en tiempo real del progreso de cada producto, mostrando las etapas completas y el estado actual del proceso.

Para desarrollar este sistema es necesario que se desarrolle con el lenguaje de programación Java ya que es un lenguaje de programación robusto, puede hacer el uso de hilos, además este lenguaje es orientado a objetos y muy bueno para aplicar las soluciones necesarias para este problema. Por lo que se requiere que cree las siguientes funcionalidades:

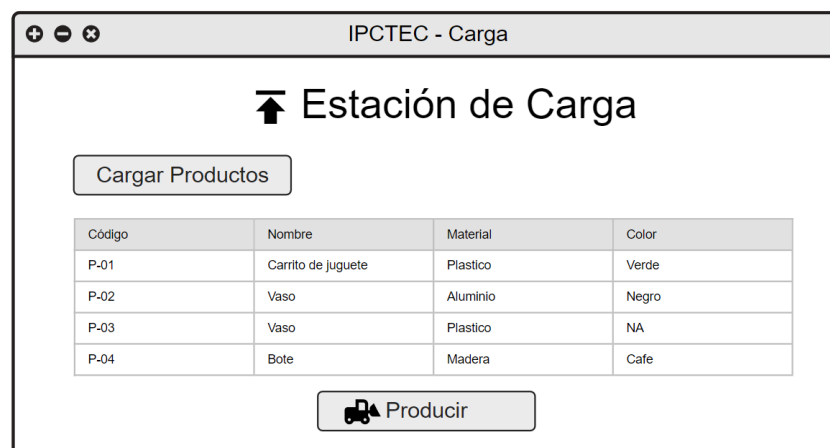
Estación de Carga

Al abrir la aplicación, esta será la primera opción a la cuál le aparecerá al usuario y consta de cargar la información de los productos que se quieren producir, cada producto cuenta con un tiempo diferente dependiendo del tipo de material y color de pintura que requieren.

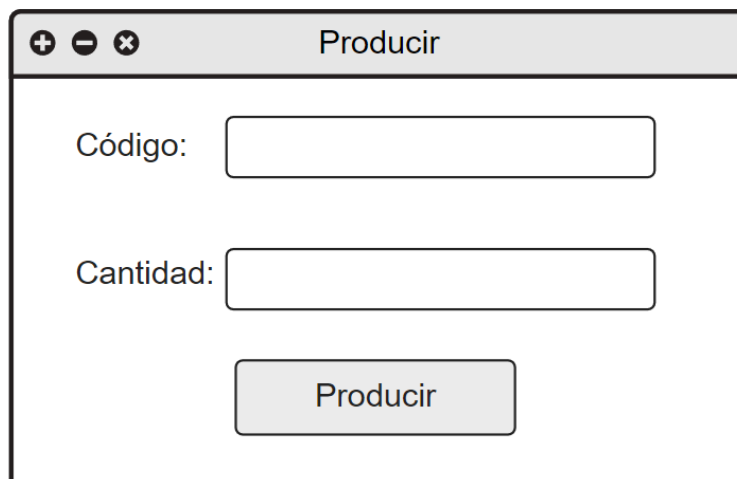
En esta estación, únicamente se requieren cargar archivos de tipo csv, los cuáles cuentan con el siguiente formato:

```
codigo, nombre producto, material, color
P-01,Carrito de juguete, plastico, verde
P-02,Vaso, aluminio, negro
P-03,Vaso, vidrio, NA
```

Además se quiere visualizar una tabla de los productos que se quieren producir.



Al darle la opción de “Producir”, le aparecerá el siguiente formulario:



El formulario tiene un título 'Producir' en la parte superior. Debajo del título, hay dos campos de entrada de texto. El primer campo está etiquetado como 'Código:' y el segundo como 'Cantidad:'. Debajo de estos campos, hay un botón rectangular con el texto 'Producir'.

En donde ingresa el código del producto y además la cantidad de productos que quiere producir.

Estación de Trabajo

Posteriormente irá a la siguiente estación de trabajo, donde se tomará en cuenta los siguientes parámetros:

FASE DE ENSAMBLAJE

Dependiendo del material eso se tardará en producir el producto:

Color	Tiempo (en s)	Costo (por segundo)
METAL	15	Q3.00
MADERA	25	Q1.00
VIDRIO	10	Q6.00
NYLON	20	Q2.00
HULE	10	Q5.00
POLIÉSTER	5	Q4.00

Terminando la fase de ensamblaje, pasará a la siguiente fase la cuál es de pintura.

FASE DE PINTURA

Dependiendo del color eso se tardará en producir el producto:

Color	Tiempo (en s)	Costo (por segundo)
VERDE	15	Q3.00
NEGRO	25	Q1.00
NA	0	Q0.00
AZUL	20	Q2.00
ROJO	10	Q5.00
AMARILLO	5	Q4.00

Terminando la fase de pintura, pasará a la siguiente fase la cuál es de empaquetado.

FASE DE EMPAQUETADO

En esta fase se realiza el empaquetado de los productos y este tiene un tiempo fijo por cada producto el cuál es:

Acción	Tiempo (en s)	Costo (por segundo)
EMPAQUETADO	10	Q2.00

Al terminar el empaquetado, tiene que mostrar la siguiente estación que es la estación de resultados.


Esta sería la interfaz propuesta para esta estación:

+

-

×

IPCTEC - Trabajo

 Estación de Trabajo

Productos 1/10 Tiempo: 00:05

Ensamblaje

50%

Pintura

0%

Empaque

0%

Estación de Resultados

Al terminar de producir todos los productos solicitados, posteriormente aparecerá un informe de resultados en la ventana donde mostrará lo siguiente:

- Cantidad de productos
- Datos del producto (código, nombre, color y material)
- Costo total de producción
- Costo por cada producto
- Tiempo total de producción de todos los productos
- Tiempo total de producción por cada producto

The screenshot shows a window titled 'IPCTEC - Trabajo'. Inside the window, the title 'Estación de Resultados' is centered at the top. Below the title, there are two columns of text. The left column lists product details: 'Producto: P-01', 'Nombre: Vaso', 'Color: Rojo', and 'Material: Vidrio'. The right column lists production statistics: 'Cantidad: 10', 'Costo total: Q20.00', 'Costo c/u: Q2.00', 'Tiempo total: 2:50', and 'Tiempo c/u: 00:17'. At the bottom center of the window, there is a button labeled 'Regresar a la estación de carga'.

Producto: P-01	Cantidad: 10
Nombre: Vaso	Costo total: Q20.00
Color: Rojo	Costo c/u: Q2.00
Material: Vidrio	Tiempo total: 2:50
	Tiempo c/u: 00:17

Regresar a la estación de carga

Librerías Permitidas

- AWT
- Swing
- LinkedList
- ArrayList
- java.io

Requerimientos

Documentación

- Manual Técnico (descripción de los métodos creados y requerimientos de la aplicación) en PDF.
- Manual de Usuario (Cómo funciona la aplicación y cómo el usuario interactúa con ella) en PDF.
- Diagrama de Clases y Diagrama de Flujo del proceso (en PDF).

Restricciones

- La aplicación debe ser desarrollada en el lenguaje de programación JAVA.
- No se permite utilizar código copiado o bajado de internet.
- El IDE por utilizar queda a discreción del estudiante (se recomienda el uso de NetBeans).
- Las copias obtendrán nota de 0 y reporte a la Escuela de Ciencias y Sistemas.
- La interfaz gráfica de usuario puede ser construida con ayuda del IDE (Drag and Drop) o con el uso de las librerías AWT y Swing.
- Durante la calificación se le solicitará al estudiante modificar el código del proyecto con el objetivo de validar la creación de este.
- Su repositorio debe estar privado.
- Cualquier librería que quiera implementar debe consultarlo primero con el auxiliar encargado del curso.
- El estudiante no tendrá derecho a calificación si no presenta interfaz gráfica, no se calificará ninguna funcionalidad en consola.
- El estudiante no tendrá derecho a calificación si no manejó su código dentro de su repositorio privado de github.
- El estudiante no tendrá derecho a calificación si no muestra sus dos hojas de calificación impresas según la fecha y hora escogida.
- Tiene que contar con mínimo 2 commits semanales y tener agregado al auxiliar de colaborador del repositorio para monitorear los avances realizados en el proyecto, de lo contrario obtendrá una penalización.

Habilidades por evaluar

- Uso de memoria estática y dinámica.
- Uso de la programación orientada a objetos.
- Habilidad para sintetizar y analizar información para solucionar problemas.
- Habilidad para comprender y realizar diagramas.
- Capacidad de crear interfaces gráficas de usuario.
- Uso de hilos.

Entrega

- **FECHA DE ENTREGA:** 18/09/2024 antes de las 23:59 (No se aceptarán entregas, ni commits a partir de esa fecha y hora).
- En su repositorio adjuntar código fuente y la documentación solicitada.
- El repositorio debe ser privado y en Github, teniendo el nombre de *IPC1_Practica1_carnet*, por ejemplo: *IPC1_Practica1_202300000*.
- Subir el enlace del repositorio de Github en la tarea asignada en UEDi.
- Agregar de colaborador del repositorio al auxiliar correspondiente a su sección antes del 11/09/2024, de lo contrario habrá penalización:

Sección	Auxiliar	Usuario de github
A	Rodrigo Alejandro Hernández de León	rodrialeipc
B	Josué Rodolfo Morales Castillo	RMorales202010033
C	David Augusto Maldonado Hurtarte	DavidMaldo02
D	Esteban Humberto Valdez Ennati	ennoxSG
E	Douglas Alexander Soch Catalán	DAlexSC
F	Ayeser Cristián Oxlay Juárez	Ayeser-Cristian
G	Federico David Zet Pajoc	fede2510