

Universidad Creativa
Q2-2021 Introducción a la Programación – Grupo 03
Escuela de informática - Tech Ground
Proyecto final: Sistema de inventarios.
Profesor: Ing. Alejandro Sánchez
Valor: 60%

Descripción: El estudiante deberá realizar tres entregables con un valor de 20% de la nota final del curso cada uno para sumar un total de 60%. Las Fechas de entrega junto con los requerimientos de cada entregable se detallarán a continuación:

Entregable 1: Jueves 10 de Junio de 2021.

Valor: 20%

Puntaje: 50 puntos.

Entregable 2: Jueves 17 de Junio de 2021.

Valor: 20%

Puntaje: 50 puntos.

Entregable 3: Jueves 24 de Junio de 2021.

Valor: 20%

Puntaje: 50 puntos.

Sistema de inventario de tienda:

Resumen:

El estudiante realizará un sistema básico de inventarios para una tienda departamental, el cual deberá ser capaz de manejar dos roles de usuario, uno administrador y otro invitado.

Además, debe ser capaz de ingresar productos al inventario, actualizarlos, eliminarlos y consultarlos (CRUD). Estos deberán estar categorizados de acuerdo con su departamento. Finalmente, estos datos deberán ser persistentes mediante el uso de archivos.

Entregable 1:

Requerimientos:

- 1- Deberá crear una “Portada” que incluya el nombre “Tienda Simón” y sus datos personales e imprimir estos datos en la terminal.
- 2- Deberá preguntar al interesado por su nombre de usuario y contraseña.
 - a. El sistema debe contar con dos roles, uno de administrador y otro de visitante.
 - b. Se recomienda el uso de diccionarios.
 - c. Cada usuario deberá contar con la siguiente información:
 - i. username
 - ii. password
 - iii. full_name
 - iv. role
 - d. Los datos para los usuarios deben ser los siguientes:
 - i. Usuario administrador:
 1. username: admin
 2. password: admin
 3. full_name: Alejandro Sanchez
 4. role: admin
 - ii. Usuario invitado:
 1. username: guest
 2. password: guest
 3. full_name: {nombre_del_estudiante}
 4. role: invitado.
- 3- Debe solicitar el nombre de usuario y contraseña.
- 4- En caso de que estos coincidan con alguno de los dos usuarios existentes, el programa debe continuar, de lo contrario debe finalizar.
- 5- Debe imprimir en pantalla: “Bienvenido al sistema de inventarios de tienda Simón, {full_name} es un placer atenderle”
- 6- Seguidamente se requiere crear un menú con las siguientes opciones si el usuario conectado es administrador:

Desea ingresar al sistema de inventarios(si/no):

-Si la respuesta es negativa, termina el programa, de lo contrario se deberá mostrar el siguiente menú:

Seleccione un departamento:

- 1- Damas
 - 2- Caballeros
 - 3- Niños

4- Salir

7- En caso de seleccionar 4, el programa finaliza.

8- Debe almacenar en memoria cual fue el departamento seleccionado y mostrar el siguiente menú:

Menú de productos del departamento {nombre_departamento}:

- 1-Consultar
- 2-Ingresa
- 3-Actualizar
- 4-Eliminar
- 5-Volver
- 6-Salir

9- En caso de seleccionar la opción 1 se debe mostrar en la pantalla “Consulta de productos”

10- En caso de seleccionar la opción 2 se debe mostrar en la pantalla “Ingresa el producto”

11- En caso de seleccionar la opción 3 se debe mostrar en la pantalla “Actualice el producto”

12- En caso de seleccionar la opción 4 se debe mostrar en la pantalla “Elimine el producto”

13- En caso de seleccionar volver, se deberá mostrar nuevamente el menú de departamentos.

14- En caso de seleccionar la opción 5 se debe mostrar en la pantalla “Saliendo del sistema de inventarios” y se debe salir del menú.

15- En caso de que el usuario conectado tenga el rol de invitado se debe mostrar el siguiente menú:

Desea ingresar al sistema de inventarios(si/no):

Seleccione un departamento:

- 1- Damas
- 2- Caballeros
- 3- Niños

Menú de productos del departamento {nombre_departamento}:

- 1-Consultar
- 2-Volver
- 3-Salir

16- En caso de seleccionar la opción 1 se debe mostrar en la pantalla “Consulta de productos”

17- En caso de seleccionar la opción 2 se debe regresar al menú de departamentos.

18- En caso de seleccionar la opción 3 se debe mostrar en la pantalla “Saliendo del sistema de inventarios” y se debe salir del menú.

Nota: Luego de completar cada acción se debe regresar al menú anterior. Por ejemplo, Luego de mostrar todos los productos se debe mostrar nuevamente el menú para seleccionar otra opción.

Entregable 2:

En este punto le agregaremos funcionalidad a nuestro sistema, almacenaremos los datos en memoria. Por lo tanto, deberá trabajar sobre el programa realizado en la entrega anterior.

Requerimientos:

- 1- Deberá crear un diccionario que contenga la siguiente información del inventario:
 - a. nombre_tienda: "Tienda Simón Escazu"
 - b. sede: "Escazu"
 - c. departamentos: [
 {nombre: "Damas",
 productos: []},
 {nombre: "Caballeros",
 productos: []},
 {nombre: "Niños",
 productos: []}]
- 2- Respetando los roles del usuario conectado deberá mostrar el menú correspondiente.
- 3- Deberá crear un contedor de productos (lista/arreglo), el cual debe contener un diccionario para cada producto el cual debe ser capaz de almacenar un código(numérico), nombre, precio unitario y cantidad para cada producto.
- 4- Como se observa en el punto 1 existe una lista de productos dentro de cada departamento.
- 5- Si el usuario selecciona la opción 1 del menú de productos del departamento, se requiere imprimir en pantalla todos productos del sistema según el departamento seleccionado.
- 6- Si el usuario selecciona la opción 2 del menú de productos del departamento, se requiere agregar un producto nuevo a la lista de productos del departamento seleccionado.
 - a. Debe solicitar al usuario el nombre del producto, código(numérico), precio unitario y cantidad en inventario.
 - b. Debe insertar este producto dentro de la lista de productos

Nota: Luego de completar cada acción se debe regresar al menú anterior. Por ejemplo, Luego de mostrar todos los productos se debe mostrar nuevamente el menú para seleccionar otra opción.

Entregable 3:

Resumen:

Este entregable depende de las entregas anteriores, su finalidad es completar la funcionalidad del sistema, agregando las funcionalidades de actualizar, eliminar productos de un departamento en específico. También se pretende poner en práctica el uso de archivos para lograr la persistencia en los datos.

Requerimientos:

- 1- En caso de que el usuario seleccione la opción 3 del menú de productos del departamento, se requiere actualizar un producto nuevo a la lista de productos del departamento seleccionado.
 - a. Debe solicitar al usuario el código del producto.
 - b. Debe buscar ese código dentro de la lista de productos del departamento seleccionado.
 - c. Al encontrar la posición de ese producto en lista debe modificar sus valores por lo tanto debe solicitar al usuario el nombre del producto, precio unitario y cantidad en inventario.
 - d. No debe modificar el código del producto.
 - e. Debe modificar el producto seleccionado con los nuevos valores.
- 2- En caso de que el usuario seleccione la opción 4 del menú de productos del departamento, se requiere eliminar un producto de acuerdo con su código. Por lo tanto, debe buscar dentro de la lista de productos del departamento y eliminar el mismo en caso de que exista, en caso contrario debe mostrar un mensaje que diga: "El producto que trata de eliminar no existe, intente nuevamente".
- 3- Debe agregar una opción adicional al menú llamada "guardar". Esta opción guardará en un archivo de texto todos los valores del diccionario de la tienda departamental.
- 4- Deberá cargar automáticamente este archivo dentro del diccionario en el sistema cada vez que el programa inicie.

Nota: Luego de completar cada acción se debe regresar al menú anterior. Por ejemplo, Luego de mostrar todos los productos se debe mostrar nuevamente el menú para seleccionar otra opción.

Rúbrica de evaluación para cada entregable:

Categoría	Excelente	Bueno	Mejorable	Deficiente	Total
Funcionamiento (50%) Indispensable para evaluar el resto de los apartados. 25pts	Que compile pero que al ejecutar haga entre el 90% y el 100% de lo que se pide. (6.25pts)	Que compile pero que al ejecutar haga entre el 70% y el 89% de lo que se pide. (6.25pts)	Que compile pero que al ejecutar haga entre el 50% y el 69% de lo que se pide. (6.25pts)	Que no compile o que al ejecutar haga menos del 49% de lo que se pide. (6.25pts)	
Claridad, legibilidad y eficiencia al programar. (20%) 10pts	El 90% y 100% es claro y legible. (2.5pts)	El 70% y 89% es claro y legible. (2.5pts)	El 50% y 69% es claro y legible. (2.5pts)	El código no es claro ni legible. (2.5pts)	
Estrategias de programación modular. (20%) 10pts	Emplea estrategias entre el 90% y 100% de las veces. (2.5pts)	Emplea estrategias entre el 70% y 89% de las veces. (2.5pts)	Emplea estrategias entre el 50% y 69% de las veces. (2.5pts)	Emplea estrategias en menos del 49%. (2.5pts)	
Variables y métodos con nombres significativos (10%) 5tps	Entre el 90% y el 100% de las variables tienen nombres representativos. (1pt)	Entre el 70% y el 89% de las variables tienen nombres representativos. (1pt)	Entre el 30% y el 69% de las variables tienen nombres representativos. (1pt)	Menos del 29% de las variables tienen nombres representativos (1pt)	

Elaborado por: Ing. Alejandro Sánchez.