Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.

Pasos de instalación:

- 1. Descargar e instalar XAMPP.
- 2. Instalar VS Code.
- 3. Descarga el archivo zip/descarga winrar.
- 4. Extrae el archivo y copia la carpeta "bsms".
- 5. Pega dentro del directorio raíz/donde instalaste XAMPP. Puede ser la unidad C, D o E. Para XAMPP, pega en: `xampp/htdocs`.
- 6. Abre PHPMyAdmin (http://localhost/phpmyadmin).
- 8. Crea una base de datos con el nombre `bsms_db.SQL`.
- 9. Importa el archivo `bsms_db.sql` (que se encuentra dentro del paquete zip en la carpeta SQL).
- 10. Ejecuta el script en http://localhost/bsms.

Webgrafía:

(https://1sourcecodr.blogspot.com)

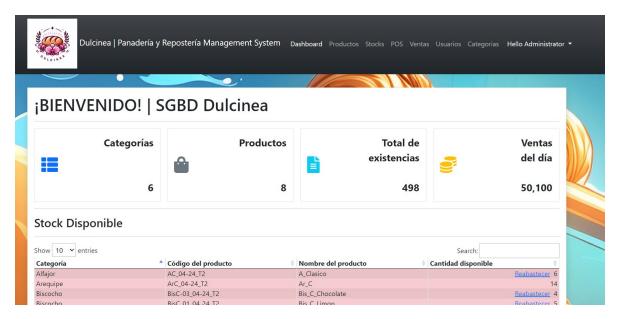
Capturas de pantalla:



Login.

Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.



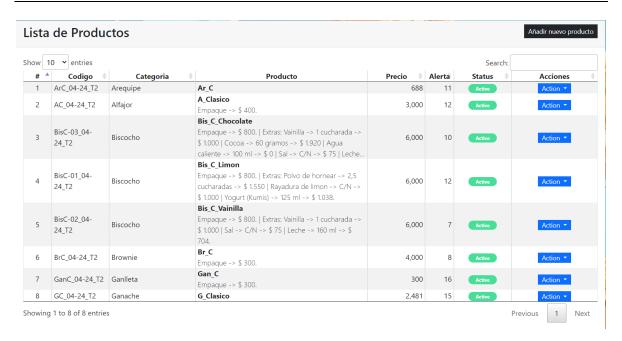
Dashboard.



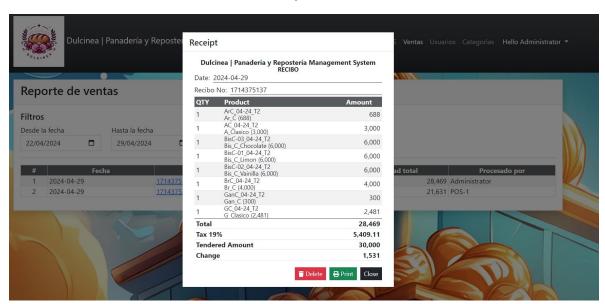
Sistema POS.

Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.



Lista de productos.



Reporte de ventas.

Credenciales:

1) Admin 1

User: admin

Password: admin123

Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.

2) Admin 2

User: DieCode24

Password: claveGenerica123

3) Cajero

User: POS-1_UTP

Password: POS-1_UTP_PASS

Mejora pedida por el profesor:

- En la sección de POS (Point of sale) se están contando los productos con cantidades menores a 1, corregirlo, ya sea por agotamiento o falta de stock.

Solución:

Codigo "Javascript" < script>:

```
$(function(){
    $('#search').on('input',function(){
    var _search = $(this).val().toLowerCase();
    $('#plist tbody tr').each(function(){
       var _text = $(this).text().toLowerCase();
       $(this).toggle(_text.includes(_search));
    });
});

$('#plist tbody tr').click(function(){
    var _tr = $(this);
    var pid = _tr.attr('data-id');
    var cname = _tr.find('.cname').text();
    var pcode = _tr.find('.pcode').text();
```

Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.

```
var name = _tr.find('.name').text();
   var price = _tr.find('.price').text().replace(/,/gi,'');
   var max = parseInt(_tr.find('.qty').text()); // Obtener la cantidad disponible
   // Agregar validación para asegurarse de que la cantidad sea al menos 1
   if (max < 1) {
     alert("Este producto está agotado o no está disponible en stock.");
     return false; // Evita agregar el producto si está agotado
   }
   var qty = 1;
   if ((\pi+\pi)).length > 0) {
     qty += parseFloat($('#item-list tbody tr[data-id="' + pid +
'"]').find('[name="quantity[]"]').val());
     if (qty > max) {
       alert("La cantidad total excede el stock disponible.");
       return false; // Evita exceder la cantidad disponible
     }
     $('#item-list tbody tr[data-id="' + pid +
'"]').find('[name="quantity[]"]').val(qty).trigger('keydown');
     return false;
   }
   var ntr = $("")
     .attr('data-id', pid)
     .append('<input class="w-100 text-center"
type="number" name="quantity[]" min="1" value="' + qty + '"/>' +
       '<input type="hidden" name="product_id[]" value="' + pid + '"/>' +
       '<input type="hidden" name="price[]" value="' + price + '"/>' +
```

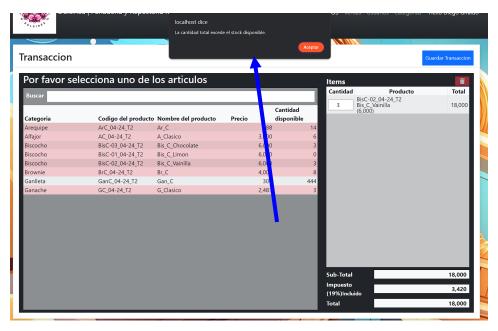
Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.

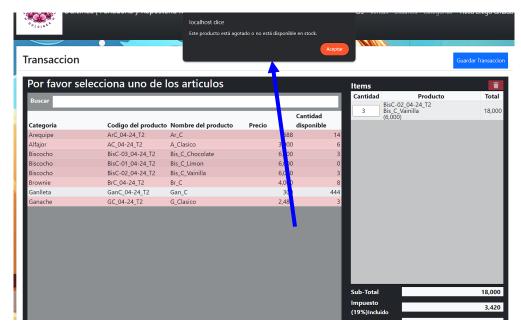
Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.

Caso 1(Agotamiento del producto en stock): en este caso es cuando hay producto disponible peor no la cantidad solicitada por el usuario.



Caso 2(Falta del producto en stock): en este caso es cuando no hay producto disponible en stock.



Elaborado por: Diego Alejandro Giraldo Ascencio.

Código: 1004754596.

Posibles y futuras mejoras:

Una posible mejora para un software de inventario y mantenimiento sería agregar una interfaz adicional dedicada a proveedores. Esta funcionalidad permitiría una mejor gestión de las relaciones con los proveedores y optimizaría la compra de ingredientes y otros recursos necesarios para la producción o el servicio. Con esta interfaz, los usuarios podrían rastrear la información de los proveedores, como contactos, historial de transacciones, tiempos de entrega y condiciones contractuales. Además, facilitaría el reabastecimiento automático de ingredientes con alertas y pedidos programados, lo que reduciría el riesgo de interrupciones en la cadena de suministro.

Otra mejora clave sería un mejor manejo de ingredientes. El software podría incorporar funcionalidades para la gestión de lotes, permitiendo a los usuarios rastrear la fecha de caducidad y la procedencia de cada lote de ingredientes. Esto no solo ayudaría a garantizar la frescura y calidad de los productos, sino también a cumplir con las regulaciones de salud y seguridad alimentaria. Asimismo, una gestión avanzada de ingredientes podría permitir el cálculo automático de costos y la previsión de requerimientos futuros, optimizando el proceso de inventario y reduciendo el desperdicio.

Además, se podría implementar una interfaz mejorada para el análisis de datos con la base de datos subyacente. Esto incluiría herramientas avanzadas de visualización y análisis que facilitarían la interpretación de grandes volúmenes de datos. Con gráficos, tablas y paneles de control intuitivos, los usuarios podrían obtener información valiosa sobre el estado actual del inventario, las tendencias de ventas y la eficiencia del mantenimiento. Una funcionalidad de este tipo permitiría a los gestores tomar decisiones informadas y estratégicas, aumentando la productividad y reduciendo los costos relacionados con el mantenimiento y el inventario.

Estas mejoras, combinadas con un enfoque centrado en el usuario y la escalabilidad del software, proporcionarían una experiencia más completa y eficiente para las empresas que buscan optimizar sus procesos de inventario y mantenimiento. Al reducir la fricción en la gestión diaria y mejorar el análisis de datos, se obtendría un sistema más confiable y versátil.