

Supermercado

Generated by Doxygen 1.9.8

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 itemcarrito Class Reference	7
4.1.1 Detailed Description	9
4.1.2 Constructor & Destructor Documentation	9
4.1.2.1 itemcarrito() [1/2]	9
4.1.2.2 itemcarrito() [2/2]	9
4.1.3 Member Function Documentation	10
4.1.3.1 agregarAlCarrito()	10
4.1.3.2 eliminarProducto()	11
4.1.3.3 getCantidad()	12
4.1.3.4 getProducto()	12
4.1.3.5 getTotal()	12
4.1.3.6 mostrarCarrito()	12
4.2 producto Class Reference	13
4.2.1 Detailed Description	15
4.2.2 Constructor & Destructor Documentation	15
4.2.2.1 producto()	15
4.2.3 Member Function Documentation	16
4.2.3.1 getNombres()	16
4.2.3.2 getPrecios()	16
4.2.3.3 mostrar()	16
4.2.4 Member Data Documentation	17
4.2.4.1 cantidad	17
4.2.4.2 nombres	17
4.2.4.3 precios	17
4.3 Usuario Class Reference	17
4.3.1 Detailed Description	18
4.3.2 Constructor & Destructor Documentation	18
4.3.2.1 Usuario()	18
4.3.3 Member Function Documentation	18
4.3.3.1 getNombre()	18
4.3.3.2 guardarCompra()	19
4.3.3.3 mostrarHistorial()	20
5 File Documentation	21

5.1 include/itemcarrito.h File Reference	21
5.1.1 Detailed Description	22
5.2 itemcarrito.h	22
5.3 include/producto.h File Reference	23
5.3.1 Detailed Description	24
5.4 producto.h	24
5.5 include/usuario.h File Reference	24
5.5.1 Detailed Description	25
5.6 usuario.h	26
5.7 src/itemcarrito.cpp File Reference	26
5.7.1 Detailed Description	27
5.8 src/main.cpp File Reference	27
5.8.1 Detailed Description	28
5.8.2 Function Documentation	28
5.8.2.1 main()	28
5.9 src/producto.cpp File Reference	29
5.9.1 Detailed Description	29
5.10 src/usuario.cpp File Reference	29
5.10.1 Detailed Description	30
Index	31

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

producto	13
itemcarrito	7
Usuario	17

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

itemcarrito	Representa un producto añadido al carrito de compras	7
producto	Clase base que representa la información general de los productos	13
Usuario	Representa un usuario o cliente del sistema de compras	17

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

include/ itemcarrito.h	Declaración de la clase itemcarrito, que representa un producto dentro del carrito de compras	21
include/ producto.h	Declaración de la clase producto, que gestiona la información base de los productos disponibles	23
include/ usuario.h	Declaración de la clase Usuario , que representa un cliente con historial de compras	24
src/ itemcarrito.cpp	Implementación de la clase itemcarrito, que gestiona los productos añadidos al carrito de compras	26
src/ main.cpp	Programa principal del sistema de supermercado ECI PA	27
src/ producto.cpp	Implementación de la clase producto, que gestiona la información base de los productos disponibles	29
src/ usuario.cpp	Implementación de la clase Usuario , que administra los datos y el historial de compras de un cliente	29

Chapter 4

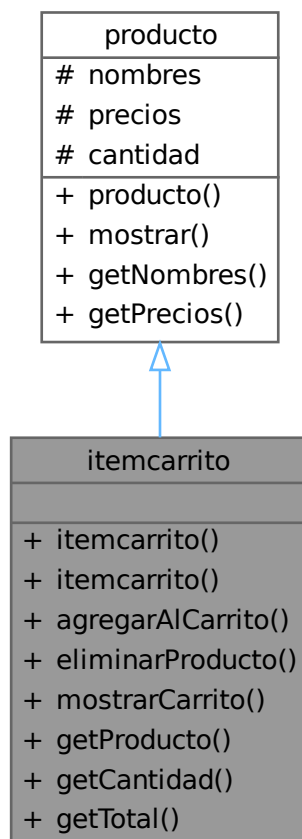
Class Documentation

4.1 itemcarrito Class Reference

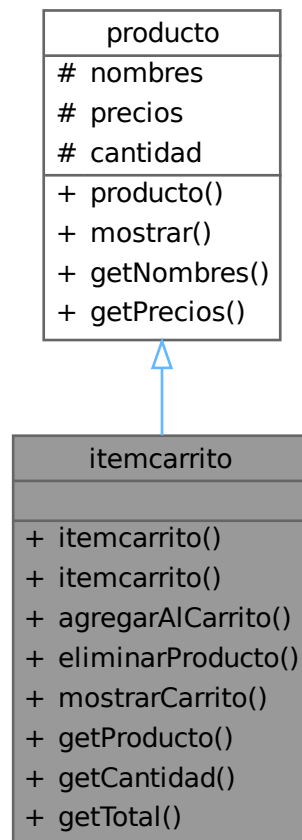
Representa un producto añadido al carrito de compras.

```
#include <itemcarrito.h>
```

Inheritance diagram for itemcarrito:



Collaboration diagram for itemcarrito:



Public Member Functions

- [itemcarrito](#) ()
Constructor por defecto.
- [itemcarrito](#) (string _producto, int _cant, int _pretot)
Constructor con parámetros.
- void [agregarAlCarrito](#) (vector< [itemcarrito](#) > &carrito)
Agrega un nuevo producto al carrito.
- void [eliminarProducto](#) (vector< [itemcarrito](#) > &carrito)
Elimina un producto del carrito.
- void [mostrarCarrito](#) (const vector< [itemcarrito](#) > &carrito)
Muestra los productos del carrito.
- string [getProducto](#) () const
Obtiene el nombre del producto.
- int [getCantidad](#) () const
Obtiene la cantidad del producto.
- int [getTotal](#) () const
Obtiene el precio total del producto.

Public Member Functions inherited from `producto`

- `producto ()`
Constructor por defecto.
- `void mostrar ()`
Muestra la lista de productos disponibles.
- `const vector< string > & getNombres () const`
Obtiene el vector de nombres de productos.
- `const vector< int > & getPrecios () const`
Obtiene el vector de precios de productos.

Additional Inherited Members

Protected Attributes inherited from `producto`

- `vector< string > nombres`
- `vector< int > precios`
- `vector< int > cantidad`

4.1.1 Detailed Description

Representa un producto añadido al carrito de compras.

Hereda de la clase `producto` y añade atributos específicos del carrito, como la cantidad de unidades y el precio total. También gestiona la manipulación de una colección de ítems.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 `itemcarrito()` [1/2]

```
itemcarrito::itemcarrito ( )
```

Constructor por defecto.

Inicializa los atributos del producto del carrito con valores predeterminados.

Inicializa los atributos del ítem con valores vacíos o en cero. Here is the caller graph for this function:



4.1.2.2 `itemcarrito()` [2/2]

```
itemcarrito::itemcarrito (
    string _producto,
    int _cant,
    int _pretot )
```

Constructor con parámetros.

Parameters

<i>_producto</i>	Nombre del producto.
<i>_cant</i>	Cantidad del producto.
<i>_pretot</i>	Precio total del producto.

Inicializa un objeto `itemcarrito` con los valores indicados.

Parameters

<i>_producto</i>	Nombre del producto.
<i>_cant</i>	Cantidad de unidades del producto.
<i>_pretot</i>	Precio total (precio unitario * cantidad).

4.1.3 Member Function Documentation

4.1.3.1 `agregarAlCarrito()`

```
void itemcarrito::agregarAlCarrito (
    vector< itemcarrito > & carrito )
```

Agrega un nuevo producto al carrito.

Agrega productos al carrito de compras.

Inserta un nuevo objeto `itemcarrito` dentro del vector de productos.

Parameters

<i>carrito</i>	Referencia al vector que almacena los ítems del carrito.
----------------	--

Permite al usuario ingresar manualmente los nombres de los productos y sus cantidades. Verifica si el producto existe en la lista heredada de `producto`, calcula el precio total y lo añade al vector de ítems del carrito.

Parameters

<i>carrito</i>	Referencia al vector de ítems donde se almacenan los productos seleccionados.
----------------	---

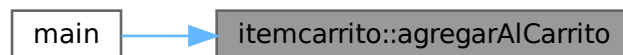
Note

Requiere que los vectores `nombres` y `precios` heredados de `producto` estén inicializados.

Here is the call graph for this function:



Here is the caller graph for this function:

**4.1.3.2 eliminarProducto()**

```
void itemcarrito::eliminarProducto (
    vector< itemcarrito > & carrito )
```

Elimina un producto del carrito.

Busca y elimina del vector el producto que coincida con el nombre almacenado en `pro`.

Parameters

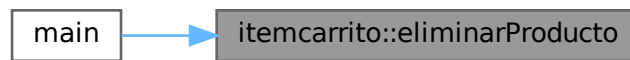
<code>carrito</code>	Referencia al vector de ítems del carrito.
----------------------	--

Solicita al usuario el nombre de un producto y, si lo encuentra en el carrito, lo elimina del vector.

Parameters

<code>carrito</code>	Referencia al vector de ítems del carrito.
----------------------	--

Here is the caller graph for this function:



4.1.3.3 getCantidad()

```
int itemcarrito::getCantidad ( ) const
```

Obtiene la cantidad del producto.

Returns

Cantidad (int).

4.1.3.4 getProducto()

```
string itemcarrito::getProducto ( ) const
```

Obtiene el nombre del producto.

Returns

Nombre del producto (string).

4.1.3.5 getTotal()

```
int itemcarrito::getTotal ( ) const
```

Obtiene el precio total del producto.

Returns

Precio total (int).

4.1.3.6 mostrarCarrito()

```
void itemcarrito::mostrarCarrito (
    const vector< itemcarrito > & carrito )
```

Muestra los productos del carrito.

Muestra el contenido actual del carrito de compras.

Imprime en consola la información de todos los productos contenidos en el vector.

Parameters

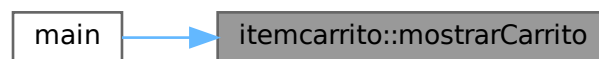
<i>carrito</i>	Vector constante con los ítems del carrito.
----------------	---

Recorre el vector de ítems e imprime los datos de cada producto: nombre, cantidad y total. También muestra el total general de la compra.

Parameters

<i>carrito</i>	Vector constante que contiene los ítems del carrito.
----------------	--

Here is the caller graph for this function:



The documentation for this class was generated from the following files:

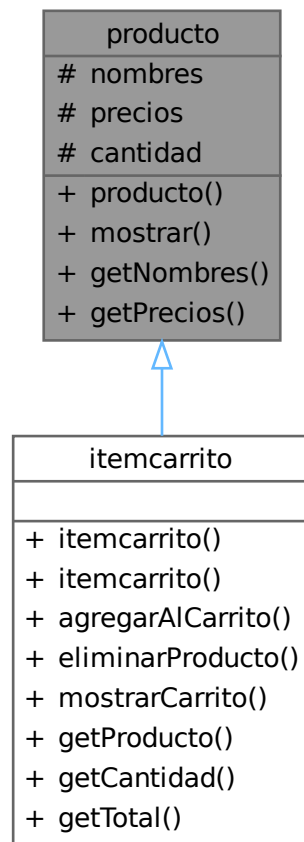
- [include/itemcarrito.h](#)
- [src/itemcarrito.cpp](#)

4.2 producto Class Reference

Clase base que representa la información general de los productos.

```
#include <producto.h>
```

Inheritance diagram for producto:



Collaboration diagram for producto:



Public Member Functions

- `producto ()`
Constructor por defecto.
- `void mostrar ()`
Muestra la lista de productos disponibles.
- `const vector< string > &getNombres () const`
Obtiene el vector de nombres de productos.
- `const vector< int > &getPrecios () const`
Obtiene el vector de precios de productos.

Protected Attributes

- `vector< string > nombres`
- `vector< int > precios`
- `vector< int > cantidad`

4.2.1 Detailed Description

Clase base que representa la información general de los productos.

Contiene vectores con los nombres, precios y cantidades de productos. Ofrece métodos para mostrar la información y acceder a los datos.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 producto()

```
producto::producto ( )
```

Constructor por defecto.

Constructor por defecto de la clase producto.

Inicializa los vectores de nombres, precios y cantidades vacíos.

Inicializa los vectores con una lista de productos predefinidos, incluyendo su nombre, precio y cantidad disponible.

Los valores por defecto son:

- arroz → \$15,000 → 5 unidades
- huevo → \$18,000 → 12 unidades
- leche → \$12,000 → 10 unidades
- pan → \$1,500 → 4 unidades
- café → \$13,000 → 2 unidades

4.2.3 Member Function Documentation

4.2.3.1 getNombres()

```
const vector< string > & producto::getNombres ( ) const
```

Obtiene el vector de nombres de productos.

Obtiene la lista de nombres de los productos.

Returns

Referencia constante al vector de nombres.

Referencia constante al vector de nombres (`vector<string>`).

4.2.3.2 getPrecios()

```
const vector< int > & producto::getPrecios ( ) const
```

Obtiene el vector de precios de productos.

Obtiene la lista de precios de los productos.

Returns

Referencia constante al vector de precios.

Referencia constante al vector de precios (`vector<int>`).

4.2.3.3 mostrar()

```
void producto::mostrar ( )
```

Muestra la lista de productos disponibles.

Imprime en consola los nombres, precios y cantidades almacenados en los vectores.

Imprime en consola el nombre, el precio y la cantidad disponible de cada producto. Here is the caller graph for this function:



4.2.4 Member Data Documentation

4.2.4.1 cantidad

```
vector<int> producto::cantidad [protected]
```

Lista de cantidades disponibles por producto.

4.2.4.2 nombres

```
vector<string> producto::nombres [protected]
```

Lista de nombres de productos disponibles.

4.2.4.3 precios

```
vector<int> producto::precios [protected]
```

Lista de precios correspondientes a cada producto.

The documentation for this class was generated from the following files:

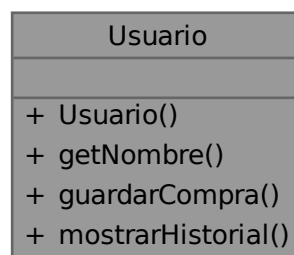
- [include/producto.h](#)
- [src/producto.cpp](#)

4.3 Usuario Class Reference

Representa un usuario o cliente del sistema de compras.

```
#include <usuario.h>
```

Collaboration diagram for Usuario:



Public Member Functions

- [Usuario](#) (string _nombre)
Constructor que inicializa el usuario con su nombre.
- string [getNombre](#) () const
Obtiene el nombre del usuario.
- void [guardarCompra](#) (const vector< [itemcarrito](#) > &carrito)
Guarda una nueva compra en el historial del usuario.
- void [mostrarHistorial](#) ()
Muestra el historial completo de compras del usuario.

4.3.1 Detailed Description

Representa un usuario o cliente del sistema de compras.

Cada usuario posee un nombre y un historial de compras. Cada compra se almacena como un vector de objetos `itemcarrito`, permitiendo registrar y mostrar las compras realizadas.

4.3.2 Constructor & Destructor Documentation

4.3.2.1 Usuario()

```
Usuario::Usuario (
    string _nombre )
```

Constructor que inicializa el usuario con su nombre.

Constructor de la clase [Usuario](#).

Parameters

<code>_nombre</code>	Nombre del usuario.
----------------------	---------------------

Inicializa el objeto con el nombre del usuario proporcionado.

Parameters

<code>_nombre</code>	Nombre del usuario o cliente.
----------------------	-------------------------------

4.3.3 Member Function Documentation

4.3.3.1 getNombre()

```
string Usuario::getNombre ( ) const
```

Obtiene el nombre del usuario.

Returns

Nombre del usuario (`string`).

Here is the caller graph for this function:

**4.3.3.2 guardarCompra()**

```
void Usuario::guardarCompra (
    const vector< itemcarrito > & carrito )
```

Guarda una nueva compra en el historial del usuario.

Guarda una compra en el historial del usuario.

Cada compra se almacena como un vector de objetos `itemcarrito`.

Parameters

<code>carrito</code>	Carrito de compra que contiene los productos adquiridos.
----------------------	--

Añade un nuevo vector de objetos `itemcarrito` al historial de compras. Cada vector representa una compra completa (carrito finalizado).

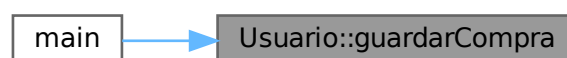
Parameters

<code>carrito</code>	Vector que contiene los productos comprados.
----------------------	--

Note

Se asume que el vector `carrito` contiene ítems válidos y con totales calculados.

Here is the caller graph for this function:



4.3.3.3 mostrarHistorial()

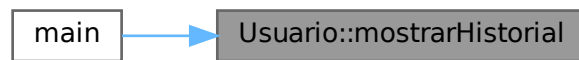
```
void Usuario::mostrarHistorial ( )
```

Muestra el historial completo de compras del usuario.

Muestra el historial de compras del usuario.

Recorre y muestra todas las compras realizadas por el usuario, junto con los productos de cada una.

Recorre todas las compras almacenadas y muestra los productos, cantidades y totales de cada una. También indica el total de cada compra. Here is the caller graph for this function:



The documentation for this class was generated from the following files:

- [include/usuario.h](#)
- [src/usuario.cpp](#)

Chapter 5

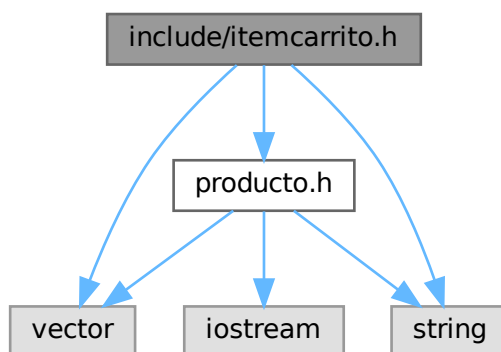
File Documentation

5.1 include/itemcarrito.h File Reference

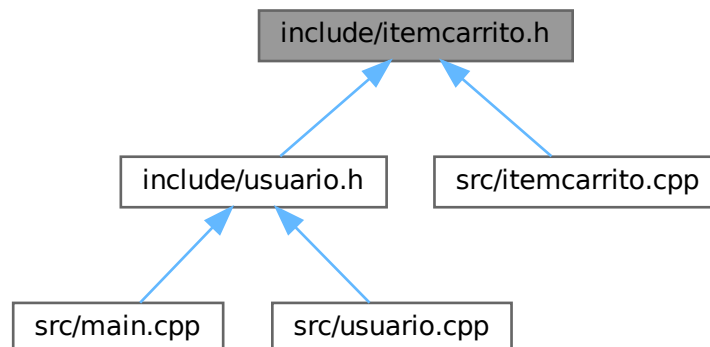
Declaración de la clase itemcarrito, que representa un producto dentro del carrito de compras.

```
#include "producto.h"  
#include <vector>  
#include <string>
```

Include dependency graph for itemcarrito.h:



This graph shows which files directly or indirectly include this file:



Classes

- class `itemcarrito`

Representa un producto añadido al carrito de compras.

5.1.1 Detailed Description

Declaración de la clase `itemcarrito`, que representa un producto dentro del carrito de compras.

Esta clase hereda de la clase `producto` y agrega información relacionada con la cantidad de productos y el precio total correspondiente a esa cantidad. Además, provee métodos para administrar un vector de ítems en el carrito, como agregar, eliminar y mostrar productos.

5.2 `itemcarrito.h`

[Go to the documentation of this file.](#)

```

00001
00011 #ifndef ITEMCARRITO_H
00012 #define ITEMCARRITO_H
00013
00014 #include "producto.h"
00015 #include <vector>
00016 #include <string>
00017 using namespace std;
00018
00027 class itemcarrito : public producto {
00028 private:
00029     string pro;
00030     int cant;
00031     int pretot;
00033 public:
00039     itemcarrito();
00040
00047     itemcarrito(string _producto, int _cant, int _pretot);
00048
00056     void agregarAlCarrito(vector<itemcarrito> &carrito);
00057
00065     void eliminarProducto(vector<itemcarrito> &carrito);
  
```

```

00066
00074     void mostrarCarrito(const vector<itemcarrito> &carrito);
00075
00080     string getProducto() const;
00081
00086     int getCantidad() const;
00087
00092     int getTotal() const;
00093 };
00094
00095 #endif // ITEMCARRITO_H

```

5.3 include/producto.h File Reference

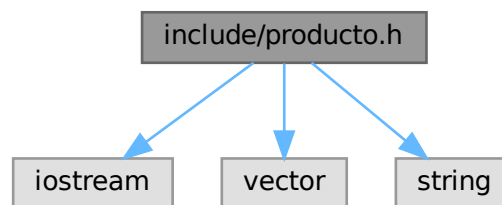
Declaración de la clase producto, que gestiona la información base de los productos disponibles.

```

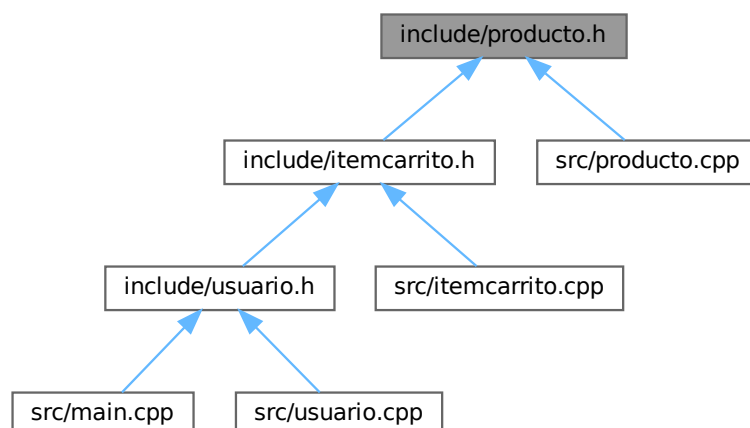
#include <iostream>
#include <vector>
#include <string>

```

Include dependency graph for producto.h:



This graph shows which files directly or indirectly include this file:



Classes

- class `producto`

Clase base que representa la información general de los productos.

5.3.1 Detailed Description

Declaración de la clase `producto`, que gestiona la información base de los productos disponibles.

La clase `producto` almacena listas (vectores) con los nombres, precios y cantidades disponibles de los productos. Sirve como clase base para `itemcarrito`, desde la cual se extienden las funcionalidades relacionadas con el carrito de compras.

5.4 `producto.h`

[Go to the documentation of this file.](#)

```
00001
00012 #ifndef PRODUCTO_H
00013 #define PRODUCTO_H
00014
00015 #include <iostream>
00016 #include <vector>
00017 #include <string>
00018 using namespace std;
00019
00027 class producto {
00028 protected:
00029     vector<string> nombres;
00030     vector<int> precios;
00031     vector<int> cantidad;
00033 public:
00039     producto();
00040
00046     void mostrar();
00047
00052     const vector<string>& getNombres() const;
00053
00058     const vector<int>& getPrecios() const;
00059 };
00060
00061 #endif // PRODUCTO_H
```

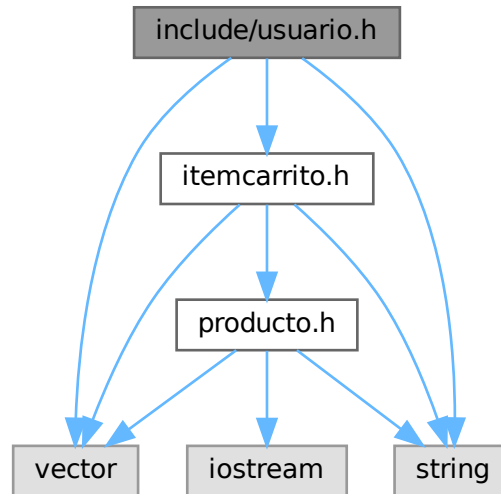
5.5 `include/usuario.h` File Reference

Declaración de la clase `Usuario`, que representa un cliente con historial de compras.

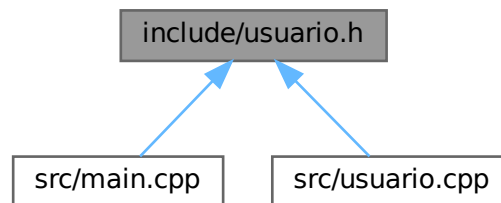
```
#include "itemcarrito.h"
#include <string>
```

```
#include <vector>
```

Include dependency graph for usuario.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Usuario](#)

Representa un usuario o cliente del sistema de compras.

5.5.1 Detailed Description

Declaración de la clase [Usuario](#), que representa un cliente con historial de compras.

La clase [Usuario](#) almacena el nombre del cliente y un historial de compras, donde cada compra se compone de un conjunto (vector) de objetos `itemcarrito`. Permite registrar nuevas compras y visualizar el historial completo.

5.6 usuario.h

[Go to the documentation of this file.](#)

```

00001
00011 #ifndef USUARIO_H
00012 #define USUARIO_H
00013
00014 #include "itemcarrito.h"
00015 #include <string>
00016 #include <vector>
00017 using namespace std;
00018
00027 class Usuario {
00028 private:
00029     string nombre;
00030     vector<vector<itemcarrito> > historialCompras;
00032 public:
00037     Usuario(string _nombre);
00038
00043     string getNombre() const;
00044
00052     void guardarCompra(const vector<itemcarrito> &carrito);
00053
00060     void mostrarHistorial();
00061 };
00062
00063 #endif // USUARIO_H
00064

```

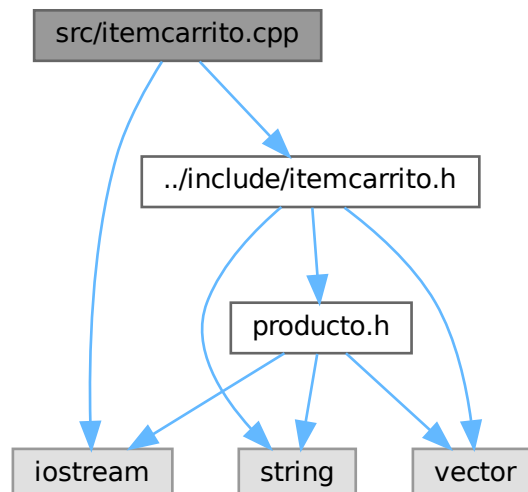
5.7 src/itemcarrito.cpp File Reference

Implementación de la clase itemcarrito, que gestiona los productos añadidos al carrito de compras.

```
#include "../include/itemcarrito.h"
```

```
#include <iostream>
```

Include dependency graph for itemcarrito.cpp:



5.7.1 Detailed Description

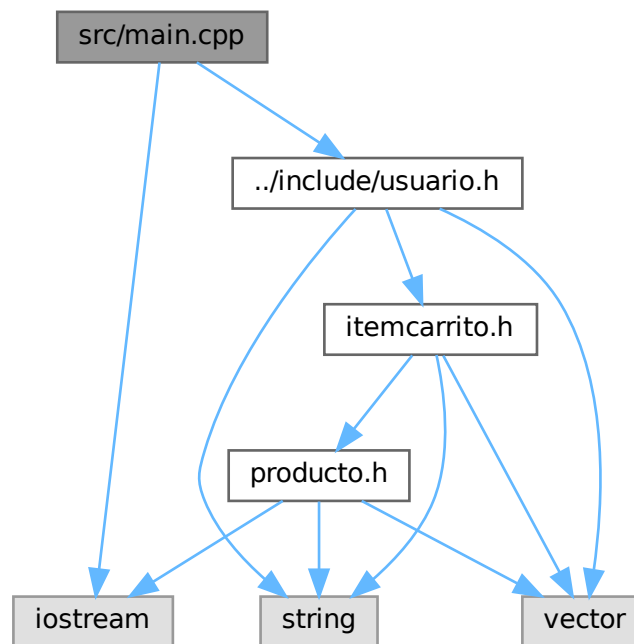
Implementación de la clase itemcarrito, que gestiona los productos añadidos al carrito de compras.

Este archivo define los métodos que permiten agregar, eliminar y mostrar los productos dentro del carrito, así como acceder a los atributos del ítem.

5.8 src/main.cpp File Reference

Programa principal del sistema de supermercado ECI PA.

```
#include "../include/usuario.h"  
#include <iostream>  
Include dependency graph for main.cpp:
```



Functions

- `int main ()`

Función principal del programa.

5.8.1 Detailed Description

Programa principal del sistema de supermercado ECI PA.

Este archivo implementa la lógica general del sistema, permitiendo:

- Registrar y gestionar múltiples usuarios.
- Visualizar la lista de productos disponibles.
- Agregar, eliminar y mostrar productos en el carrito de compras.
- Guardar y visualizar el historial de compras de cada usuario.

Combina las clases `producto`, `itemcarrito` y `Usuario` para crear una simulación sencilla de un sistema de ventas en consola.

5.8.2 Function Documentation

5.8.2.1 `main()`

```
int main ( )
```

Función principal del programa.

Controla el menú general de la aplicación y las operaciones principales: registro de usuarios, compras, historial y visualización de productos.

Returns

0 al finalizar la ejecución correctamente.

< Lista de usuarios registrados.

< Catálogo de productos disponibles.

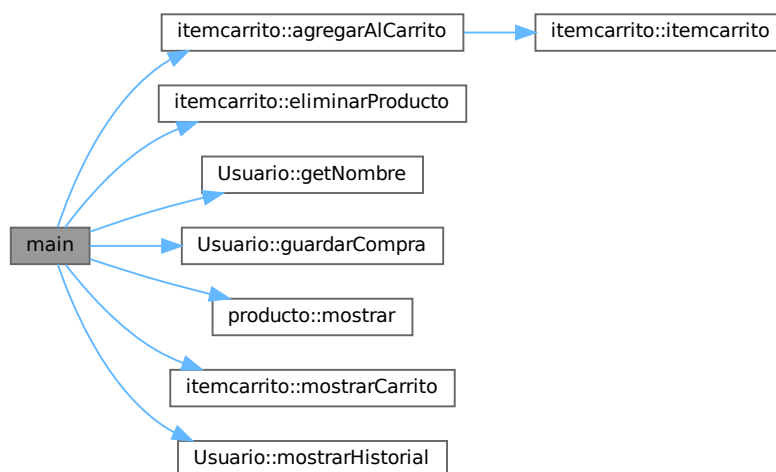
< Objeto auxiliar para manipular el carrito.

< Carrito de compras temporal del usuario.

< Nombre del usuario actual.

< Índice del usuario seleccionado.

< Opción elegida en el menú principal. Here is the call graph for this function:



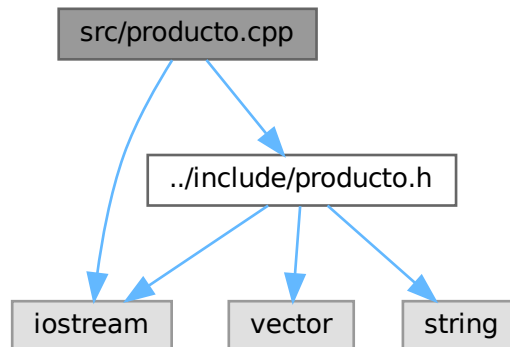
5.9 src/producto.cpp File Reference

Implementación de la clase producto, que gestiona la información base de los productos disponibles.

```
#include "../include/producto.h"
```

```
#include <iostream>
```

Include dependency graph for producto.cpp:



5.9.1 Detailed Description

Implementación de la clase producto, que gestiona la información base de los productos disponibles.

Este archivo define los métodos de la clase `producto`, encargada de inicializar, mostrar y proveer acceso a los datos de productos.

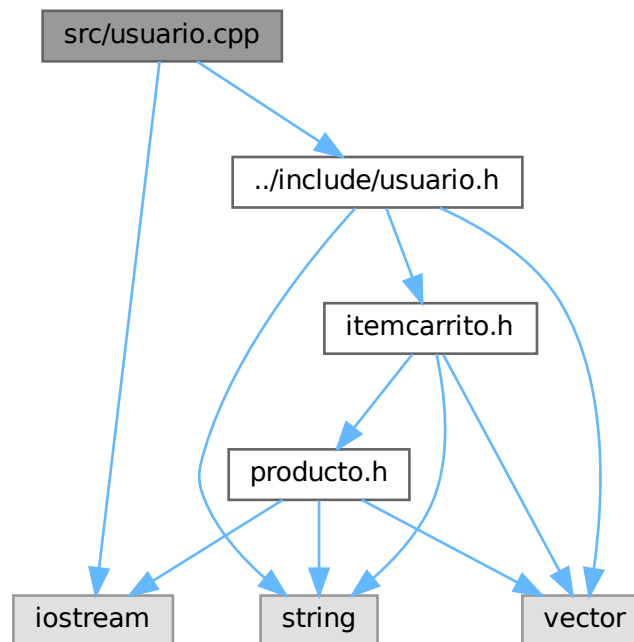
5.10 src/usuario.cpp File Reference

Implementación de la clase [Usuario](#), que administra los datos y el historial de compras de un cliente.

```
#include "../include/usuario.h"
```

```
#include <iostream>
```

Include dependency graph for `usuario.cpp`:



5.10.1 Detailed Description

Implementación de la clase `Usuario`, que administra los datos y el historial de compras de un cliente.

Este archivo define los métodos de la clase `Usuario`, encargada de manejar el nombre del cliente, almacenar sus compras y mostrar el historial completo.

Index

agregarAlCarrito
 itemcarrito, [10](#)

cantidad
 producto, [17](#)

eliminarProducto
 itemcarrito, [11](#)

getCantidad
 itemcarrito, [12](#)

getNombre
 Usuario, [18](#)

getNombres
 producto, [16](#)

getPrecios
 producto, [16](#)

getProducto
 itemcarrito, [12](#)

getTotal
 itemcarrito, [12](#)

guardarCompra
 Usuario, [19](#)

include/itemcarrito.h, [21](#), [22](#)

include/producto.h, [23](#), [24](#)

include/usuario.h, [24](#), [26](#)

itemcarrito, [7](#)
 agregarAlCarrito, [10](#)
 eliminarProducto, [11](#)
 getCantidad, [12](#)
 getProducto, [12](#)
 getTotal, [12](#)
 itemcarrito, [9](#)
 mostrarCarrito, [12](#)

main
 main.cpp, [28](#)

main.cpp
 main, [28](#)

mostrar
 producto, [16](#)

mostrarCarrito
 itemcarrito, [12](#)

mostrarHistorial
 Usuario, [20](#)

nombres
 producto, [17](#)

precios

 producto, [17](#)

producto, [13](#)
 cantidad, [17](#)
 getNombres, [16](#)
 getPrecios, [16](#)
 mostrar, [16](#)
 nombres, [17](#)
 precios, [17](#)
 producto, [15](#)

src/itemcarrito.cpp, [26](#)

src/main.cpp, [27](#)

src/producto.cpp, [29](#)

src/usuario.cpp, [29](#)

Usuario, [17](#)
 getNombre, [18](#)
 guardarCompra, [19](#)
 mostrarHistorial, [20](#)
 Usuario, [18](#)