API DOCUMENTATION

# WEB API

1. The WebAPI documentation is found in the default page when running the API/Api.sln solution.
2. Open the page using a browser on <http://localhost:1947>

# CLIENT LIBRARY

* 1. To use the client library the ClientLibrary dll must be referenced in the application that wishes to consume it. The nuget package for Newtonsoft.json **must also be** installed in the consumer application.
  2. The consumer application should have a static instance of the Orders class instantiated once, the base url can be changed using the constructor, the default test url is localhost:1947

static Orders myOrders = new Orders("http://localhost:1947");

* 1. The methods in the ClientLibrary are async, and may be called using Task.Run(..) as shown below:

Task.Run(async () =>

{

bool added = await myOrders.Add(new OrderItem() {productid = 1, quantity=1}) .GetAwaiter().GetResult();

}

* 1. Client Library accessible Methods for Orders Class:
     + 1. async Task<List<OrderItem>> GetAll()

Description:

Allows the client to fetch a list of orders placed.

Returns:

List<OrderItem> containing orders that were made. **An exception is thrown by the method in case of failure.**

* + - 1. async Task<OrderItem> GetOrderItem(int **productid**)

Description:

Allows the client to fetch details of a single order item.

Parameters:

**productid** – the product id of the OrderItem to be retrieved

Returns:

OrderItem for the **productid** specified. **An exception is thrown by the method in case of failure.**

* + - 1. async Task<bool> Add(OrderItem orderItem)

Description:

This method allows the client to add an Order. If the product already exists in the current order, the quantity is incremented.

Parameters:

**orderItem** – an OrderItem for the order to be added

Return:

Boolean value True if Order was succesfully added

* + - 1. async Task<bool> Update(ulong productid, OrderItem orderItem)

Description:

Allows the client to update details of a single order item. If the product already exists in the current order, the quantity is overwritten.

Parameters:

**productid** – the product id of the OrderItem to be updated

**orderItem** – an OrderItem with updated quantity value.

Returns:

Boolean value True if Order was succesfully updated

* + - 1. async Task<bool> Remove(int productid)

Description:

Allows the client to remove a single order item.

Parameters:

**productid** – the product id of the OrderItem to be removed

Returns:

Boolean value True if Order was succesfully removed

* + - 1. async Task<bool> Clear()

Description:

Allows the client to clear his Order.

Returns:

Boolean value True if the Order was succesfully cleared.

Note: the OrderItem class is available under the ClientLibrary namespace.

OrderItem contains the following properties:

ulong productid;

int quantity;