**UNIVERSIDAD DE POLITECNICA DE PACHUCA**



**Presentan:**

Carlos Enrique Chavero Brizuela

Luis Emmanuel García Ostria

Rodolfo Ian Escalante López

**MATERIA:**

Arquitecturas Orientadas a Servicios

**PROFESOR:**

Rodríguez Flores Jazmín

Zempoala, Hidalgo. 12 de diciembre de 2018

**Restaurante/ Bar “Monkys” Cd. Sahagún, Hidalgo.**

***Requerimientos Funcionales.***

* Visualizar, aprobar o rechazar facturas de los proveedores
* Ordenar compras de productos.
* Información de ingresos y egresos del bar.
* Visualización de fechas de pagos diarios y semanales a proveedores.
* Montos de pago de remuneración.
* En relación al stock de la bodega el sistema generara un stock mínimo de materias primas, y cuando la cantidad este por debajo de esa cantidad emitirá una alarma al gerente, activando la decisión de compra.
* El Gerente tendrá acceso al sistema a través de un login, el sistema le permitirá acceso a toda la información consolidada de una forma simple y clara, tendrá acceso a informes históricos.

***Requerimientos No Funcionales.***

* El software estará integrado con una base de datos que pueda ser actualizada y modificada para nuevas o futuras necesidades.
* El software deberá tener una interfaz gráfica de fácil uso e intuitiva, además debe permitir la rapidez de operación.
* Debe contar con un sistema de respaldo de información que garantice la pronta y efectiva recuperación de la información.
* El software debe permitir modificaciones y actualizaciones.
* El proveedor del software deberá entregar toda la información referente a: documentos de información de diseño, programación y uso del software, además de código fuente.

|  |  |  |
| --- | --- | --- |
| **Numero:** 1 | **Usuario:** Cliente | |
| **Nombre Historia:** Visualizar, aprobar o rechazar facturas de los proveedores | | |
| **Prioridad en negocio:**  Medio | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como gerente quiero poder visualizar mis facturas, para poder aprobar o rechazar según mi conveniencia y así poder llevar un mejor control en las facturas de mi negocio. | | |
| **Validación:** | | |

|  |  |  |
| --- | --- | --- |
| **Numero:** 2 | **Usuario:** Cliente | |
| **Nombre Historia:** Ordenar compras de productos. | | |
| **Prioridad en negocio:**  Medio | | **Riesgo en desarrollo:**  Bajo |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como gerente quiero visualizar las compras en mi negocio de forma ordenada para llevar un mejor control de las compras. | | |
| **Validación:** | | |

|  |  |  |
| --- | --- | --- |
| **Numero:** 3 | **Usuario:** Cliente | |
| **Nombre Historia:** Información de ingresos y egresos del bar. | | |
| **Prioridad en negocio:**  Alto | | **Riesgo en desarrollo:**  Alto |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como gerente quiero tener acceso a la información de egresos e ingresos del bar para tener una mejor administración de los recursos financieros. | | |

|  |  |  |
| --- | --- | --- |
| **Numero:** 4 | **Usuario:** Cliente | |
| **Nombre Historia:** Visualización de fechas de pagos diarios y semanales a proveedores. | | |
| **Prioridad en negocio:**  Medio | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como gerente quiero ver las fechas próximas de pagos para llevar un control y hacer los pagos a tiempo. | | |
| **Validación:** | | |

|  |  |  |
| --- | --- | --- |
| **Numero:** 5 | **Usuario:** Cliente | |
| **Nombre Historia:** generar un stock mínimo de materias primas | | |
| **Prioridad en negocio:**  Alto | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como gerente quiero que el sistema me alerte cuando mis materias primas lleguen a cierto número para poder reabastecerme. | | |
| **Validación:** | | |

|  |  |  |
| --- | --- | --- |
| **Numero:** 6 | **Usuario:** Cliente | |
| **Nombre Historia:** El Gerente tendrá acceso al sistema a través de un login. | | |
| **Prioridad en negocio:**  Alto | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como gerente quiero tener acceso a toda la información y al historial, para tener una mejor visión del crecimiento de mi negocio. | | |
| **Validación:** | | |

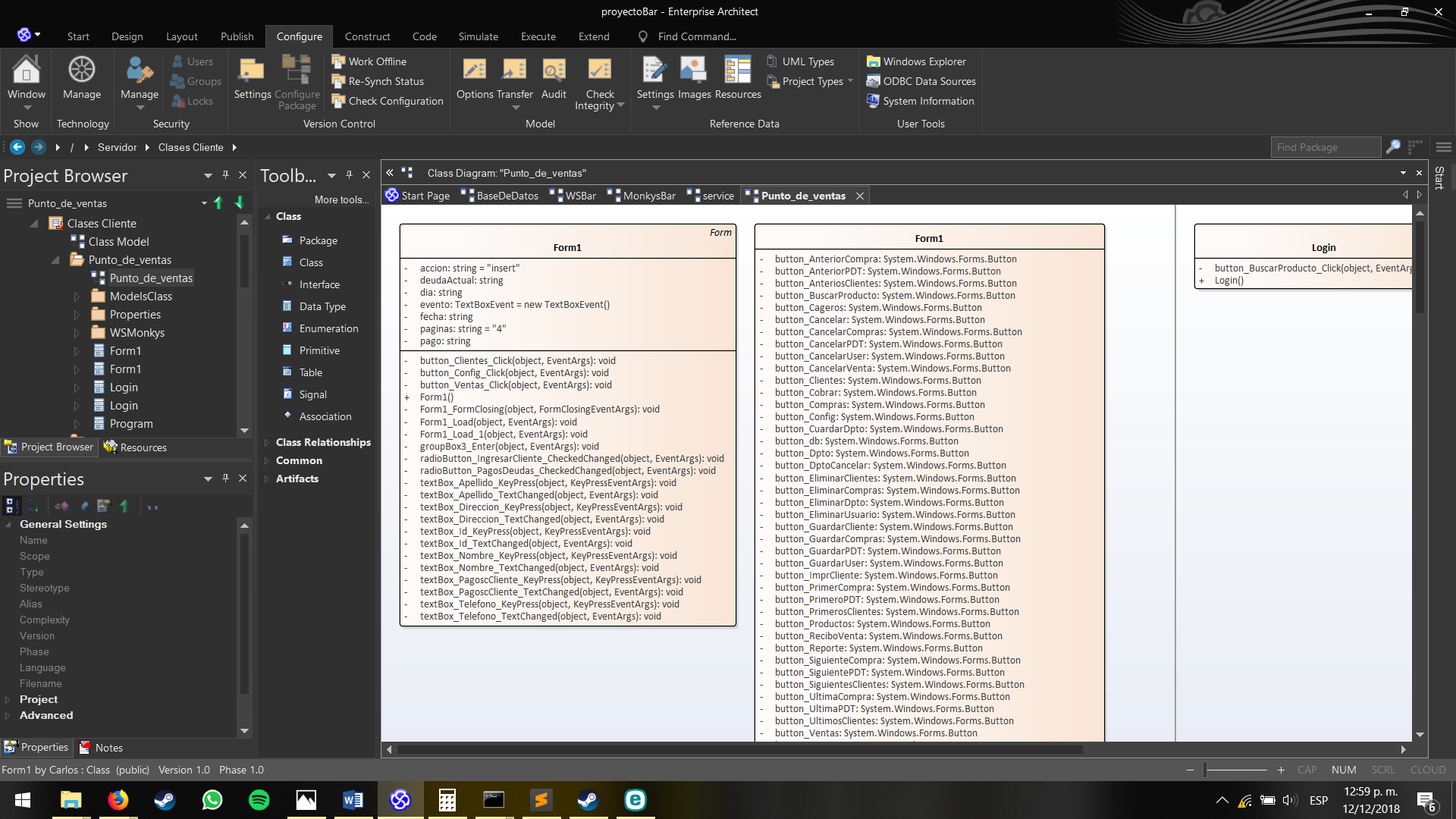
|  |  |  |
| --- | --- | --- |
| **Numero:** 7 | **Usuario:** Cliente | |
| **Nombre Historia:** Base de datos que pueda ser modificada según las necesidades del cliente. | | |
| **Prioridad en negocio:**  Alto | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como usuario quiero que el software cuente con una base de datos que pueda modificar o actualizar según las necesidades que surjan dentro del negocio. | | |
| **Validación:** | | |

|  |  |  |
| --- | --- | --- |
| **Numero:** 8 | **Usuario:** Cliente | |
| **Nombre Historia:** Interfaz rápida e intuitiva | | |
| **Prioridad en negocio:**  Medio | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como usuario quiero que el software tenga una interfaz sencilla, que no sea difícil de operar para mí y mis empleados y que incluya los logotipos de mi negocio. | | |
| **Validación:** | | |

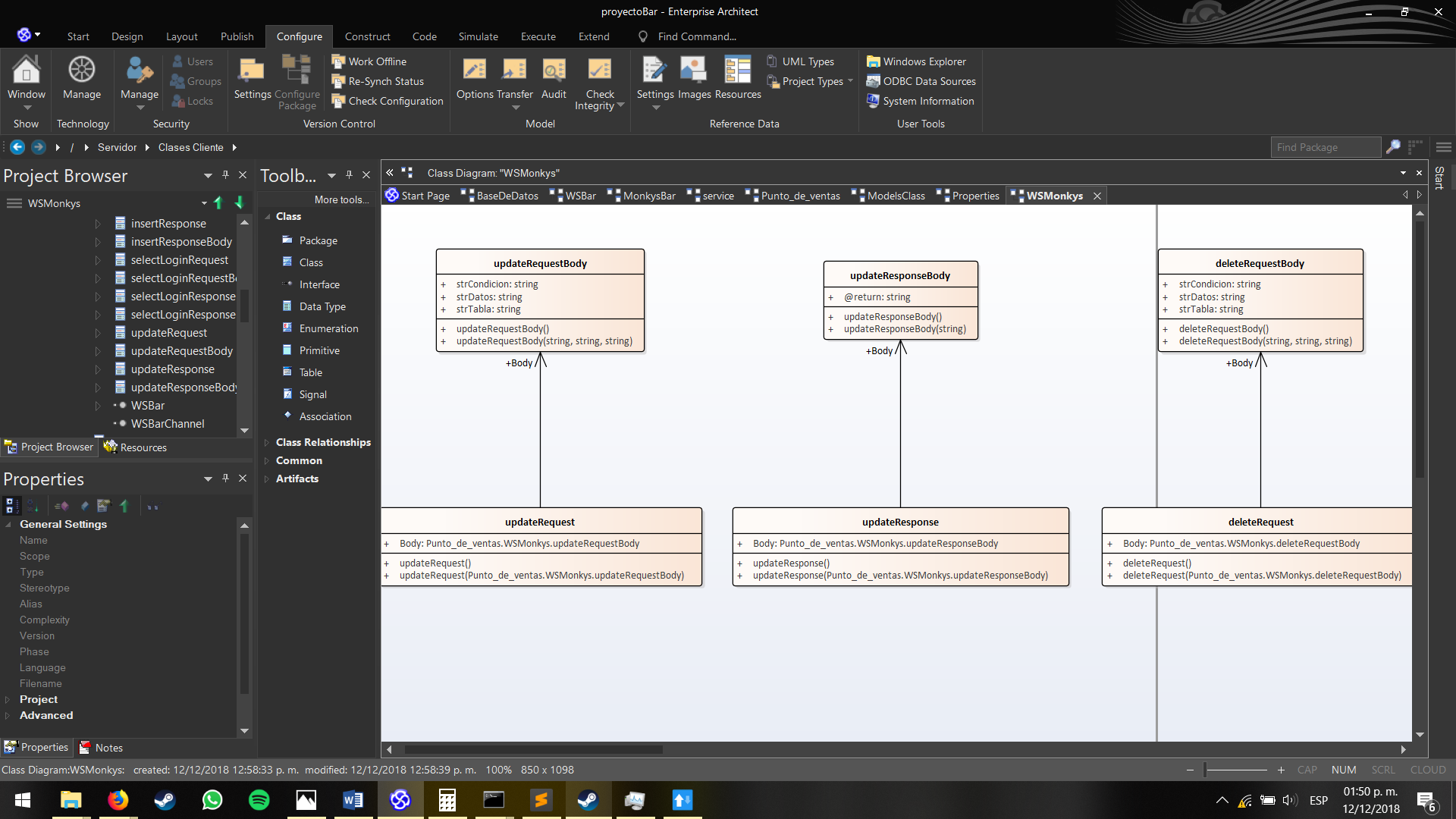
|  |  |  |
| --- | --- | --- |
| **Numero:** 9 | **Usuario:** Cliente | |
| **Nombre Historia:** Sistema de respaldo | | |
| **Prioridad en negocio:**  Alto | | **Riesgo en desarrollo:**  Alto |
| **Puntos estimados:** | | **Iteración asignada:** |
| **Programador responsable: Emmanuel Ostria, Carlos Chavero.** | | |
| **Descripción:**  Como usuario quiero que el software tenga un sistema de respaldo para que en caso de un desastre y pueda recuperar mi información con facilidad. | | |
| **Validación:** | | |

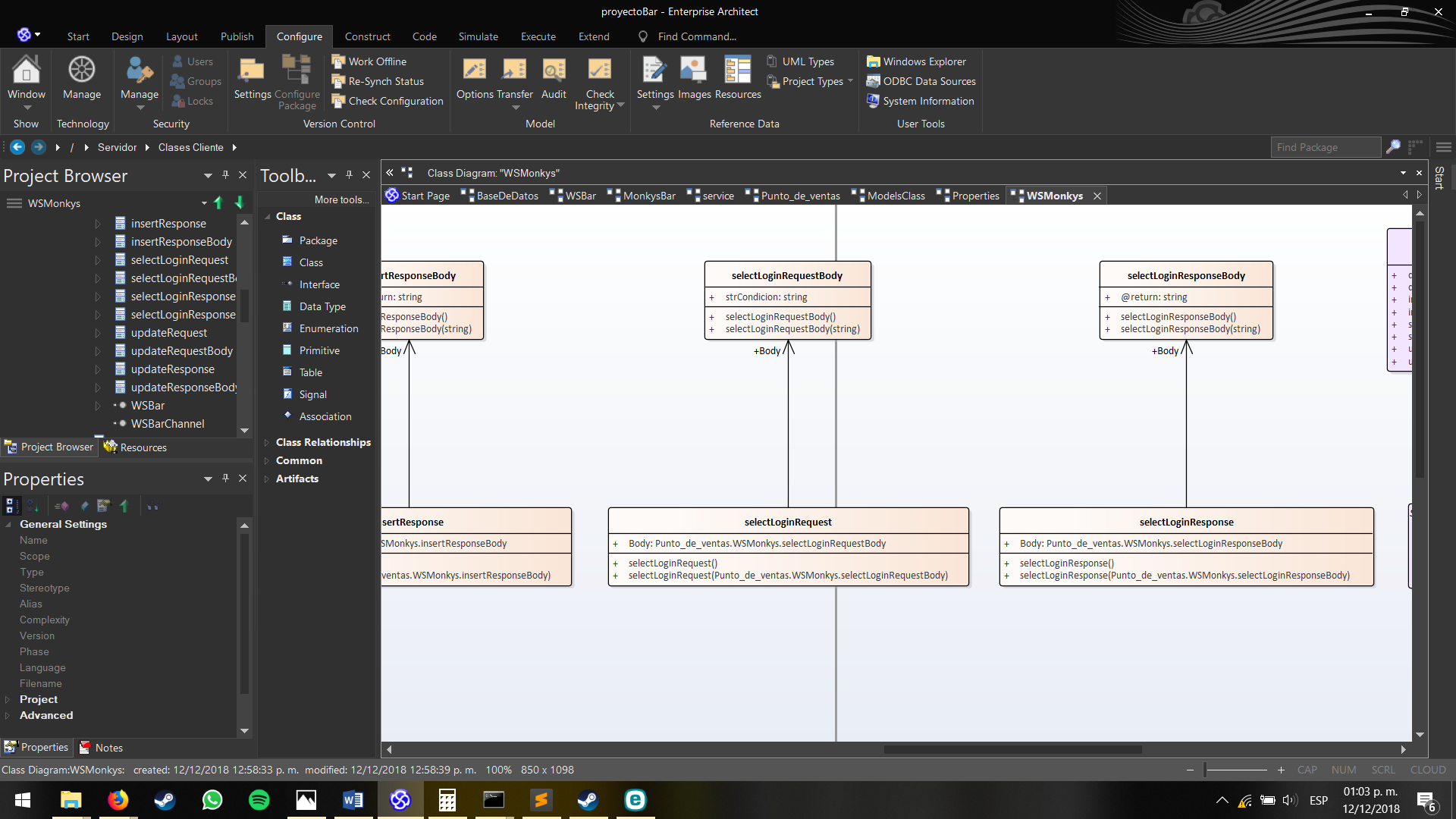
***Clases del cliente:***

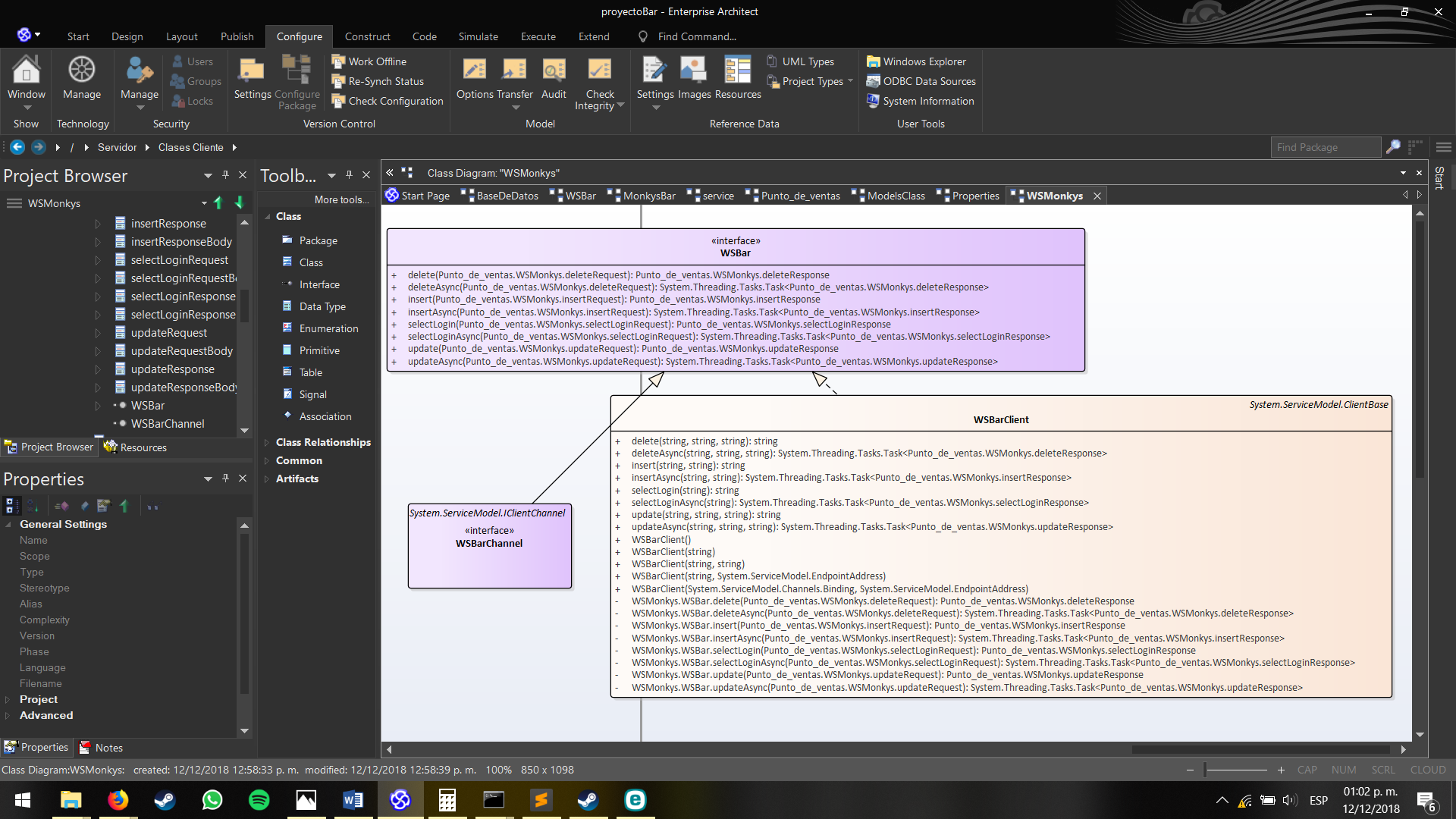
***PuntoVenta***.**Form1**



***PuntoVenta***.**WSMonkys:**

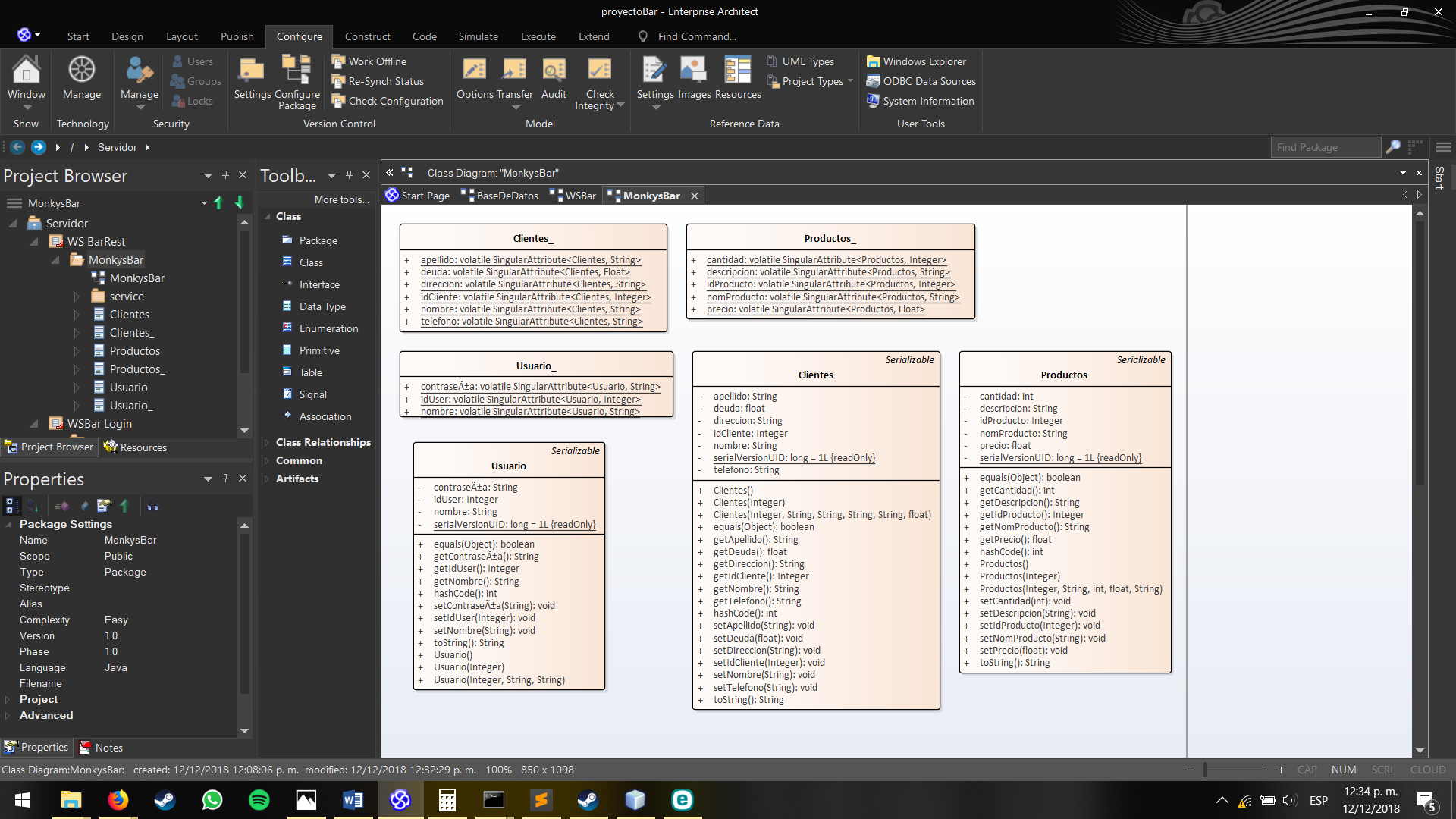




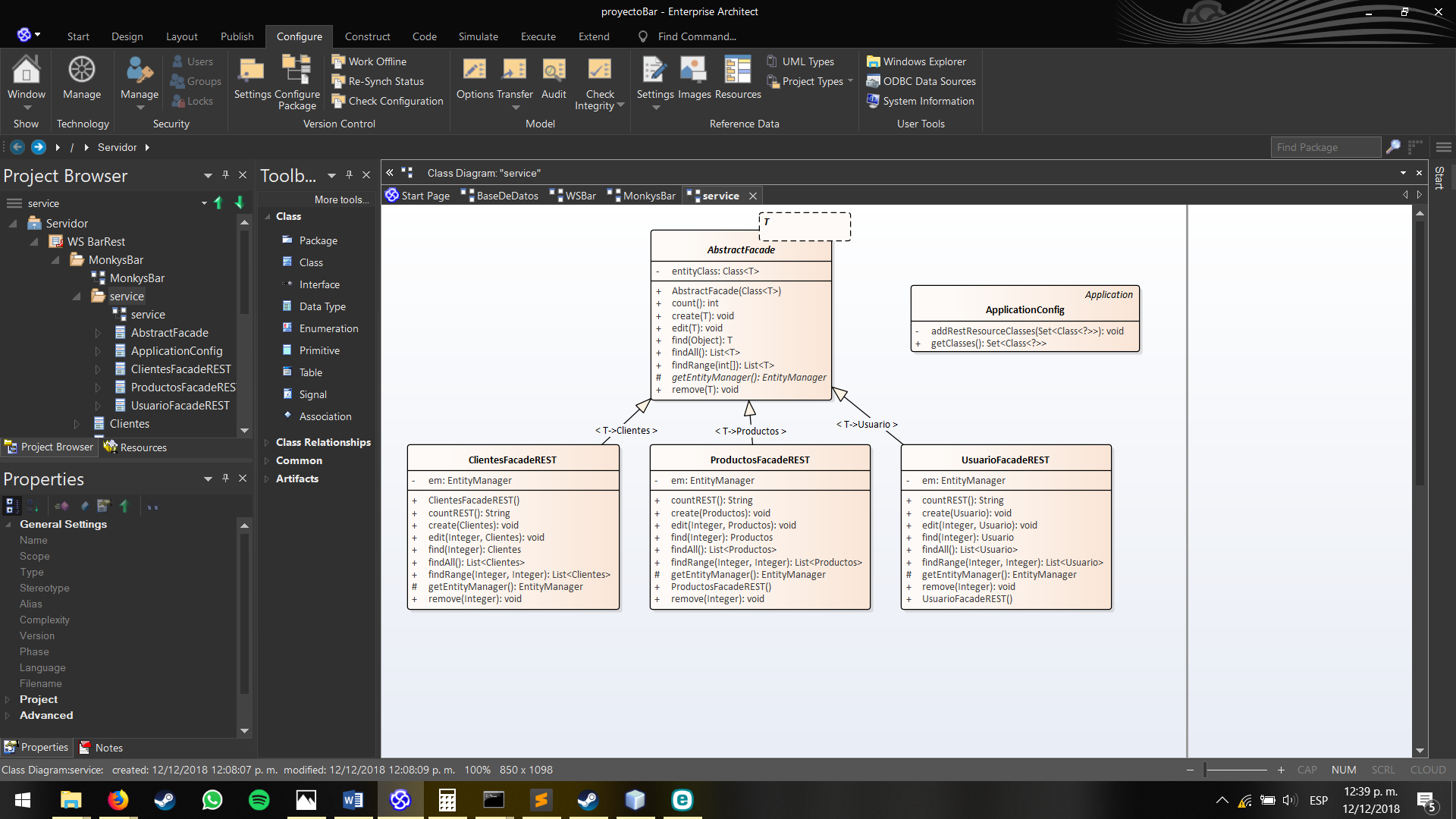


***Clases del servidor:***

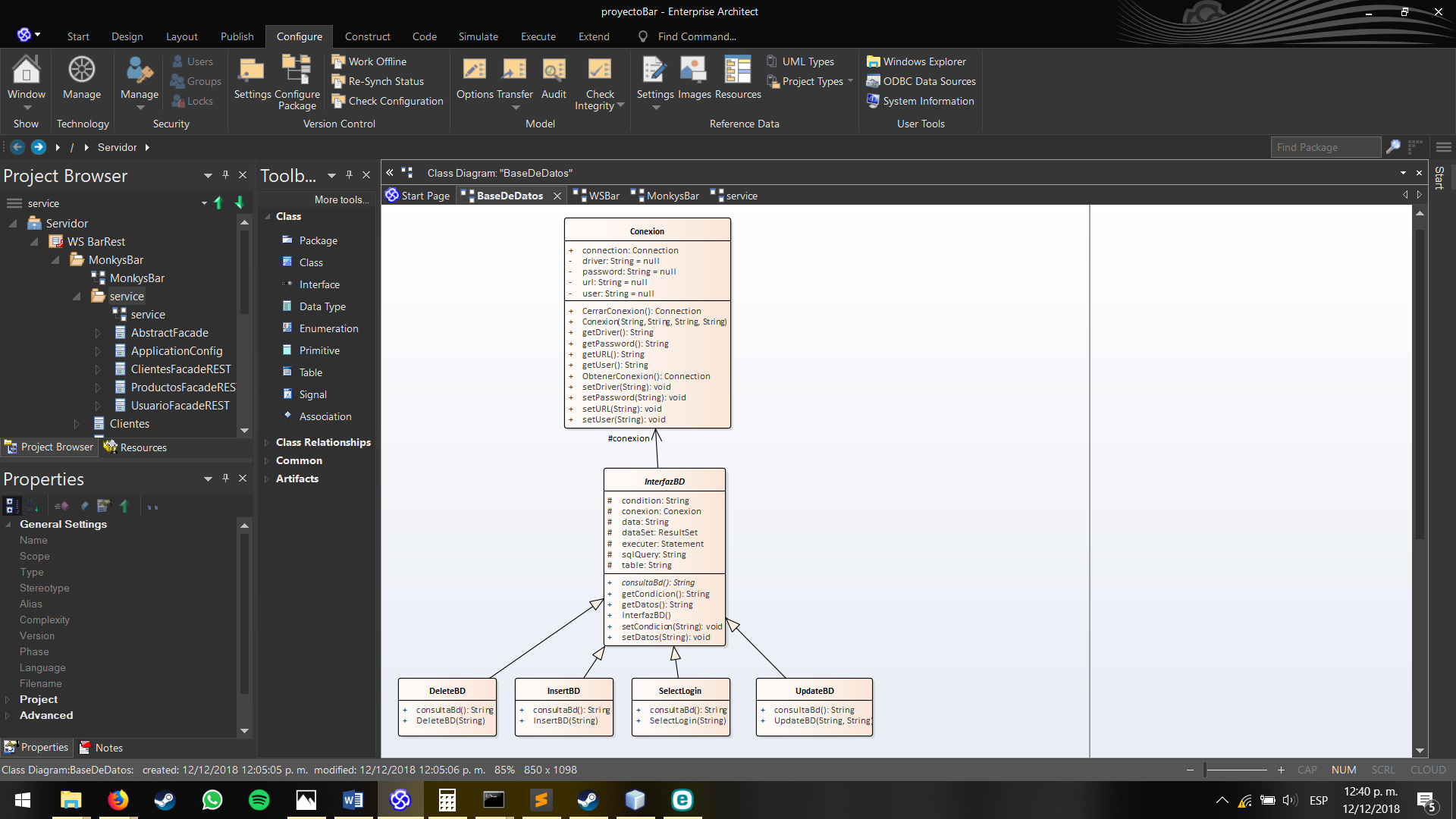
***Paquete Monkys***



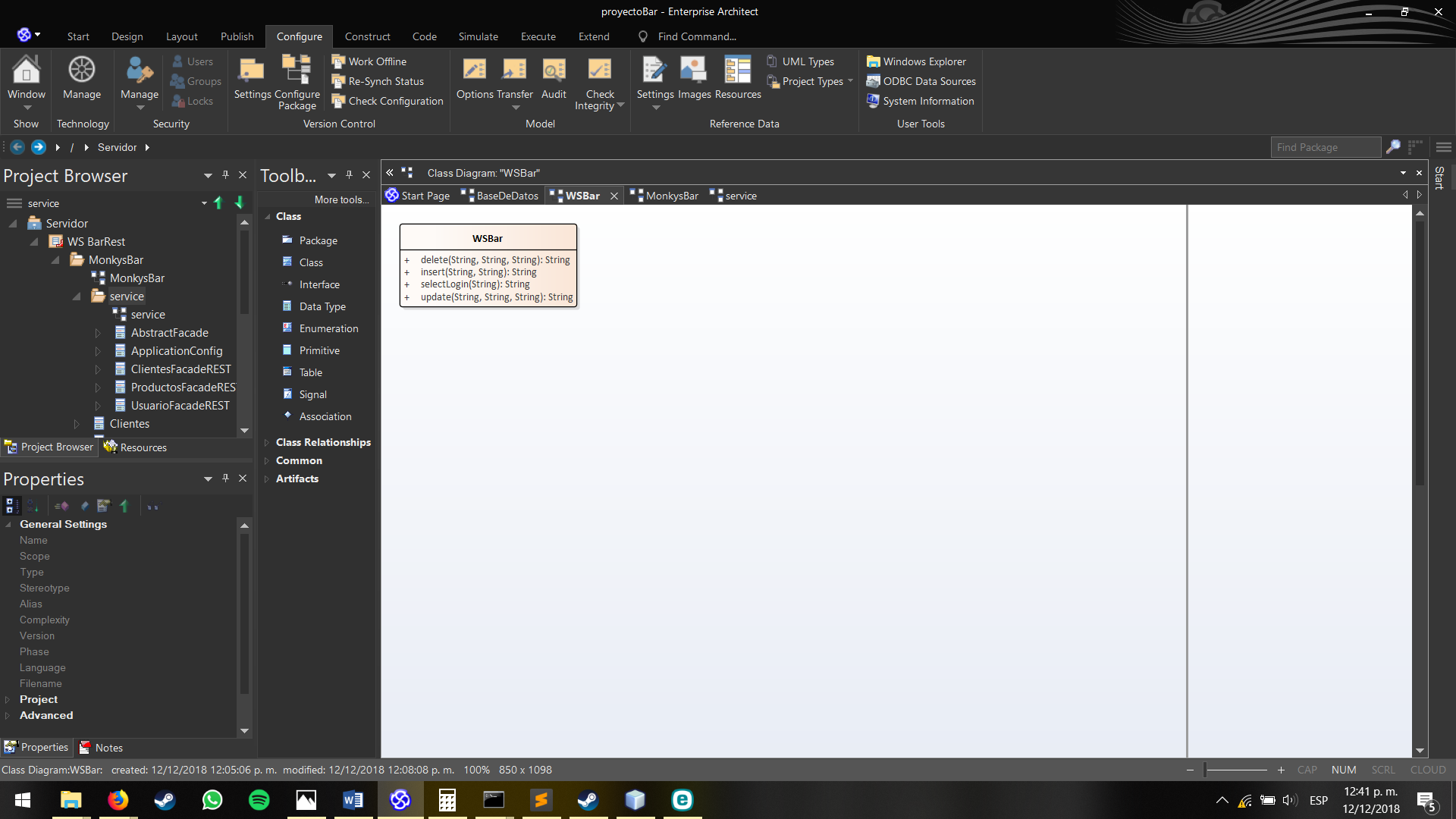
***Paquete Monkys.service***



***Paquete BaseDeDatos:***



***Paquete WSBar:***



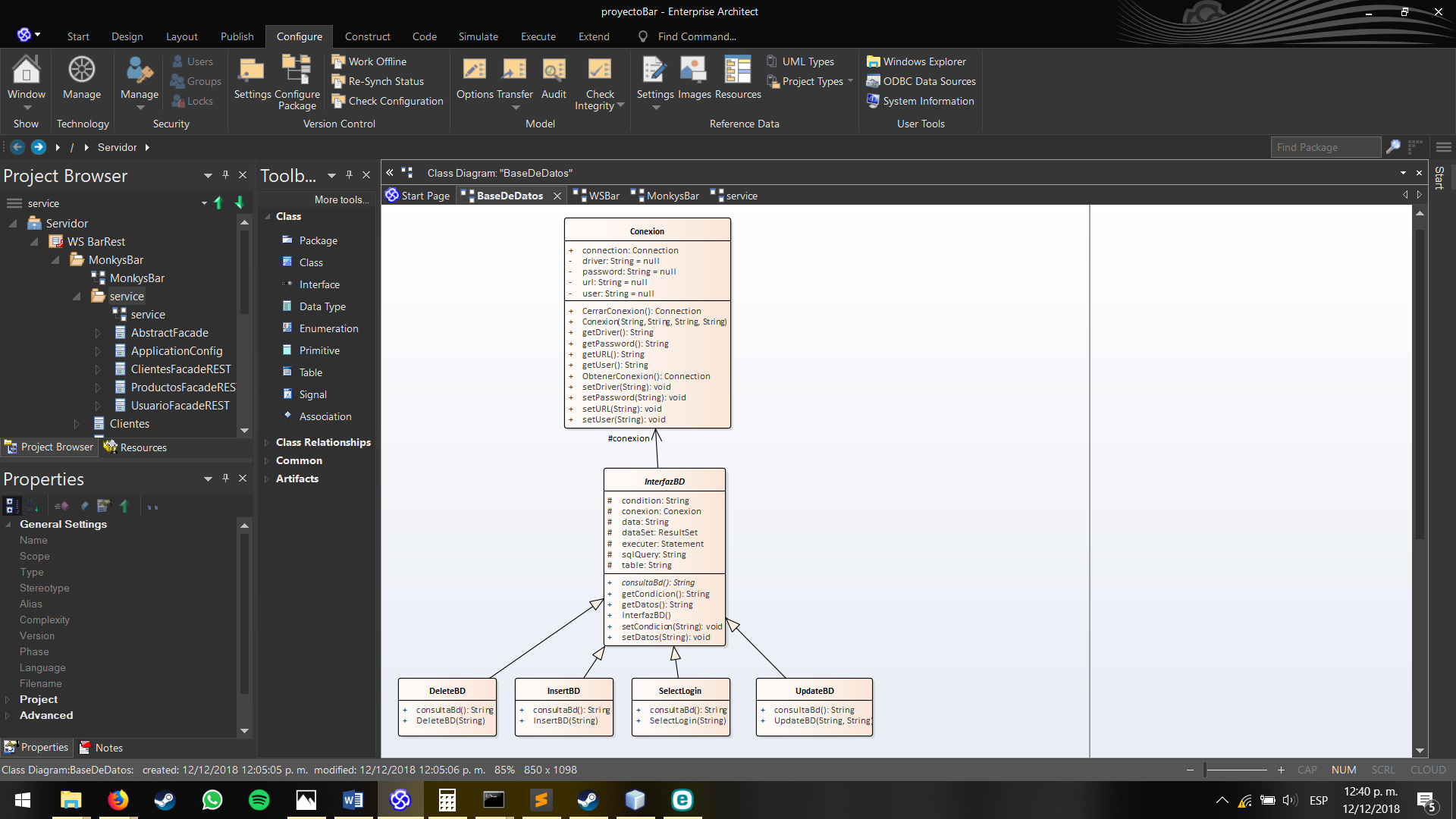
***Principios de diseño utilizados en el servidor:***

El siguiente diagrama servirá como ejemplo para demostrar la aplicación de los principios de diseño dentro de las clases del servidor.

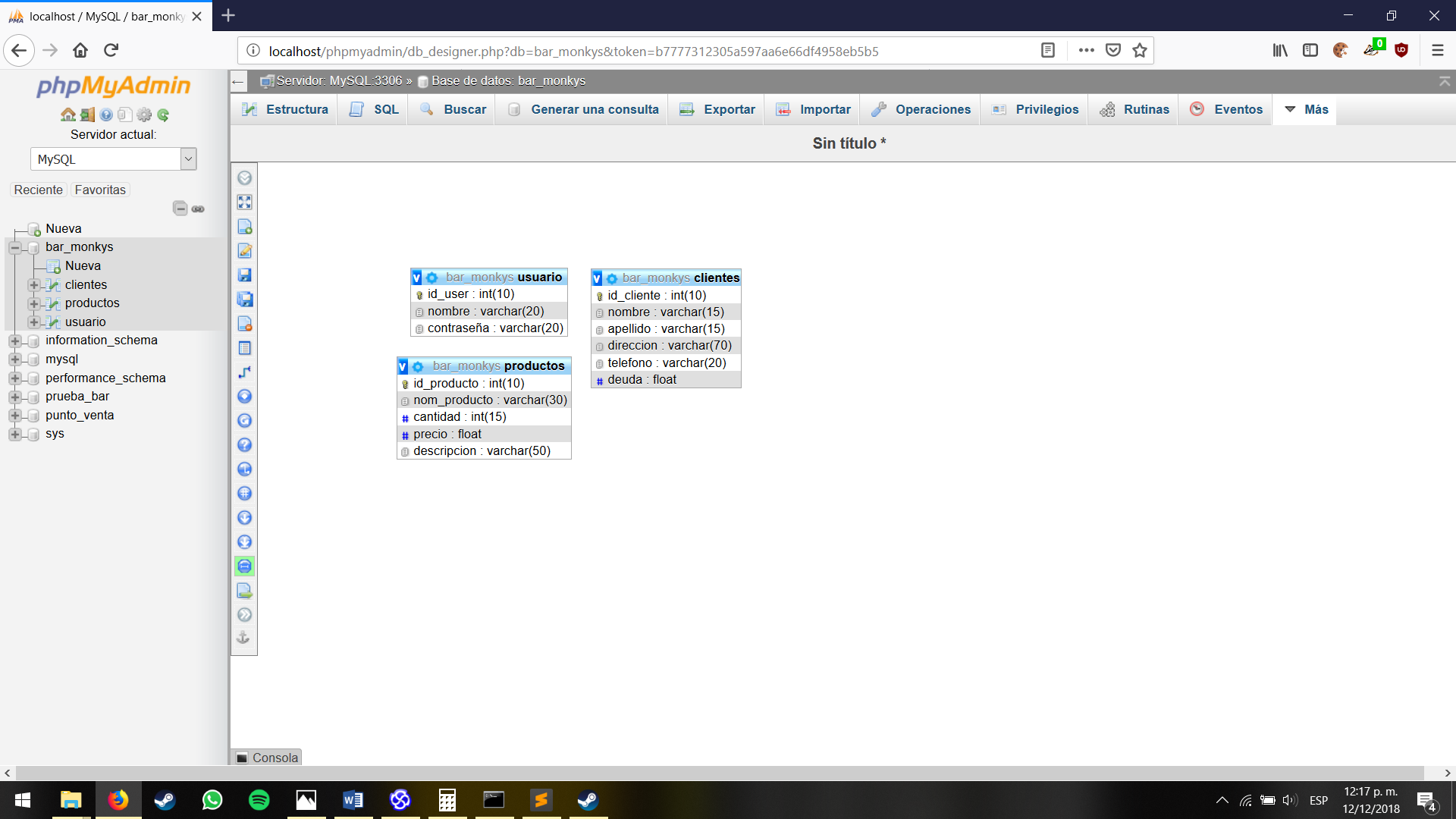
Como podemos observar, se dividieron las clases hijas en cada función básica de un *CRUD* (crear, leer, actualizar y borrar), por lo que cada una realizará solamente su tarea asignada. Por ello, podemos decir que fue utilizado el principio de **Responsabilidad única**.

También, en vez de realizar cada una la tarea de conexión, buscar la base de datos, ingresar los datos de acceso y un largo etcétera, se asignó una interfaz abstracta vinculada a la clase *conexión*, y únicamente pasará los parámetros a las clases hijas para que realicen su funcionalidad. A esto se le conoce como principio de **Segregación de interfaces**.

Por último, al estar estructurada de manera que cada clase realice únicamente la tarea asignada, podemos decir que pueden modificarse y cambiarse sin alterar al resto, por lo que se utilizó el principio **Abierto-Cerrado**.



***Modelo de base de datos:***



***Plan de Pruebas (Servidor).***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| BR | TR | Caso de prueba# | Caso de prueba | Pasos | Datos de Prueba | Resultado |
| B1 | T1 | 1 | Ver de manera remota el contenido de la tabla usuarios. | 1. Abrir advanced Rest. 2. Insertar el Request URL 3. Seleccionar la opción GET 4. Clic en SEND |  | Muestra los campos ID, nombre y contraseña de la tabla usuarios. |
| B1 | T2 | 2 | Modificar un usuario. | 1. Abrir advanced Rest. 2. Insertar el Request URL 3. Seleccionar la opción PUT 4. Clic en SEND | {  "contraseña": "1234",  "idUser": "3",  "nombre": "Carl"  } | Modifica el campo nombre de la tabla usuarios. |
| B1 | T4 | 3 | Insertar un usuario | 1. Abrir advanced Rest. 2. Insertar el Request URL 3. Seleccionar la opción POST 4. Clic en SEND | {  "contraseña": "1234",  "idUser": "2",  "nombre": "Carlos"  } | Inserta un nuevo usuario con el ID: 2, contraseña: 1234 y con el nombre de Carlos de manera exitosa en la tabla usuarios. |
| B1 | T3 | 4 | Eliminar un usuario | 1. Abrir advanced Rest. 2. Insertar el Request URL 3. Seleccionar la opción DELETE 4. Clic en SEND | {  "contraseña": "1234",  "idUser": "3",  "nombre": "Carl"  } | Elimina el registro de manera exitosa. |

**Requerimientos técnicos:**

**T1.** Puede ver el contenido de la base de datos.

**T2.** Puede modificar los datos en la base de datos.

**T3.**  Puede eliminar los datos en la base de datos.

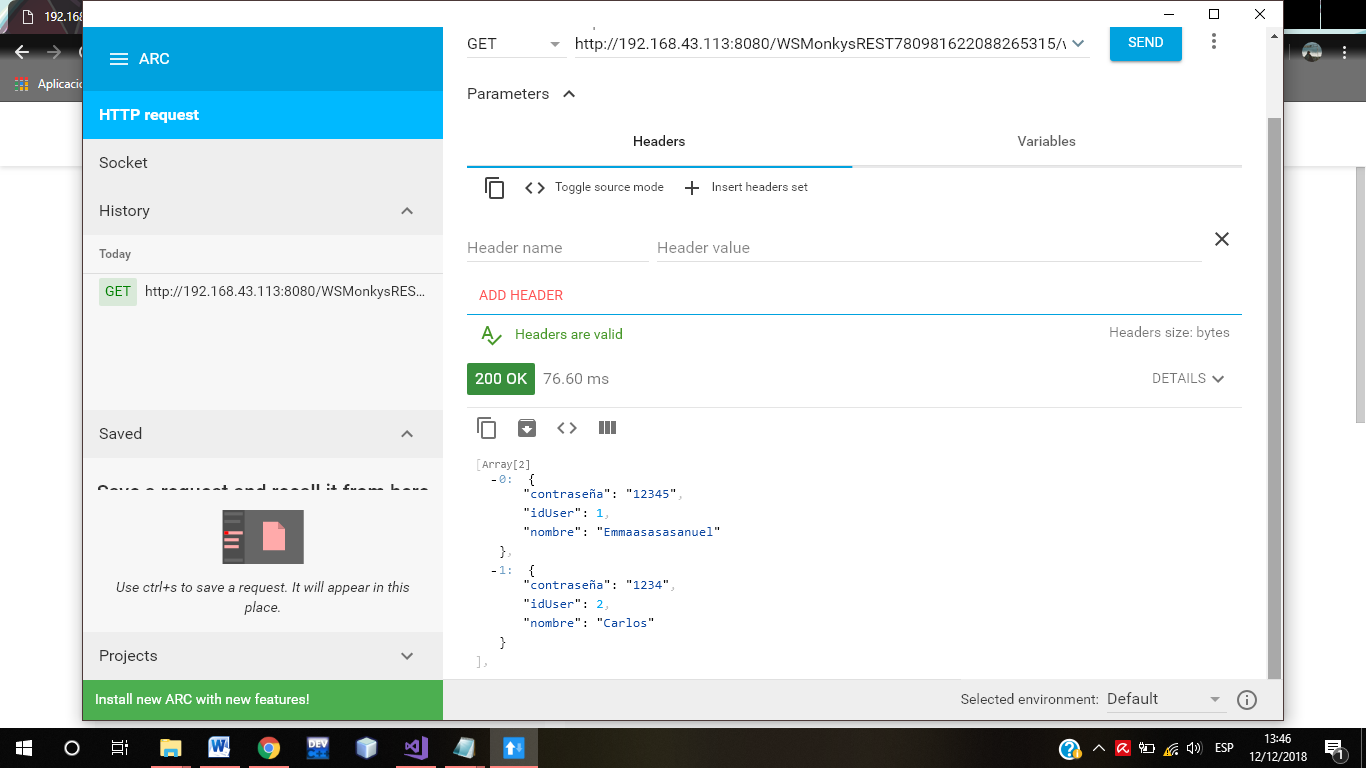
**T4.** Puede insertar los datos en la base de datos.

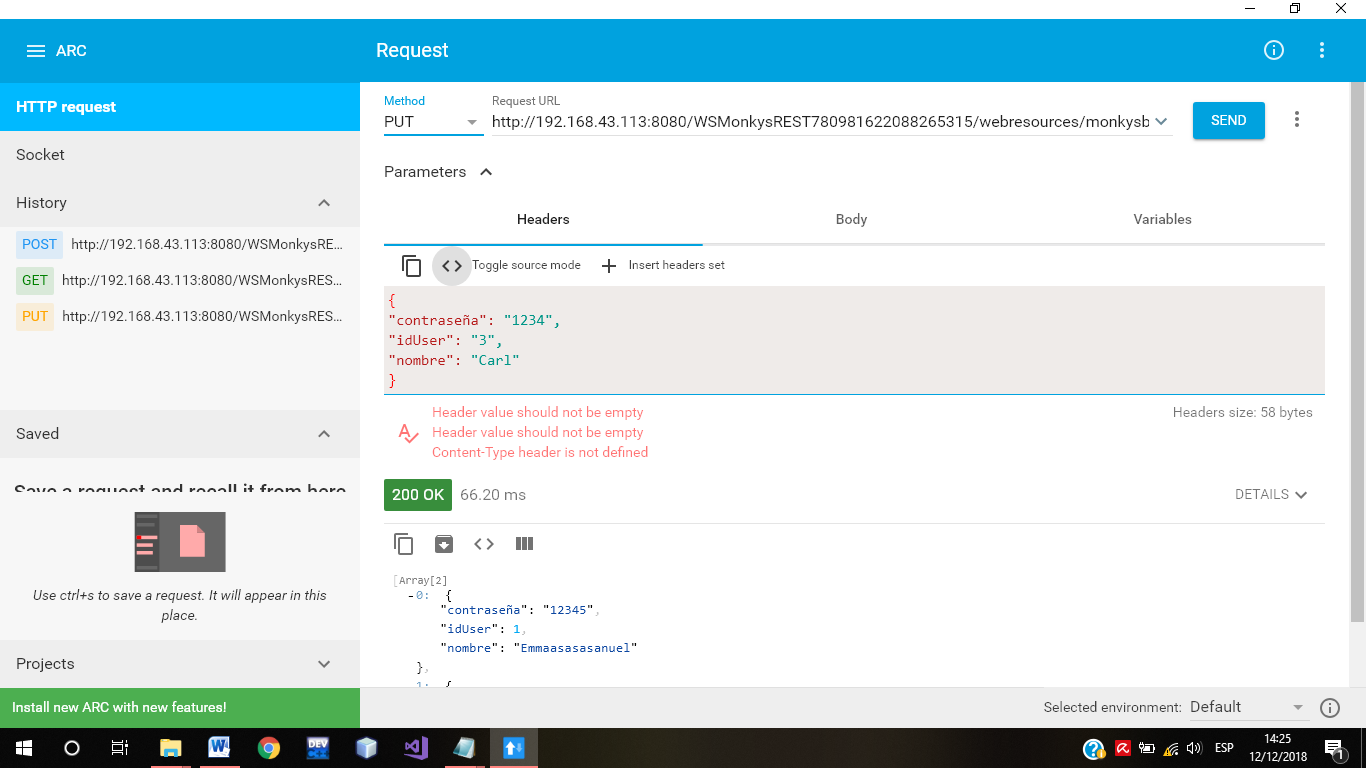
**Requerimientos de negocio:**

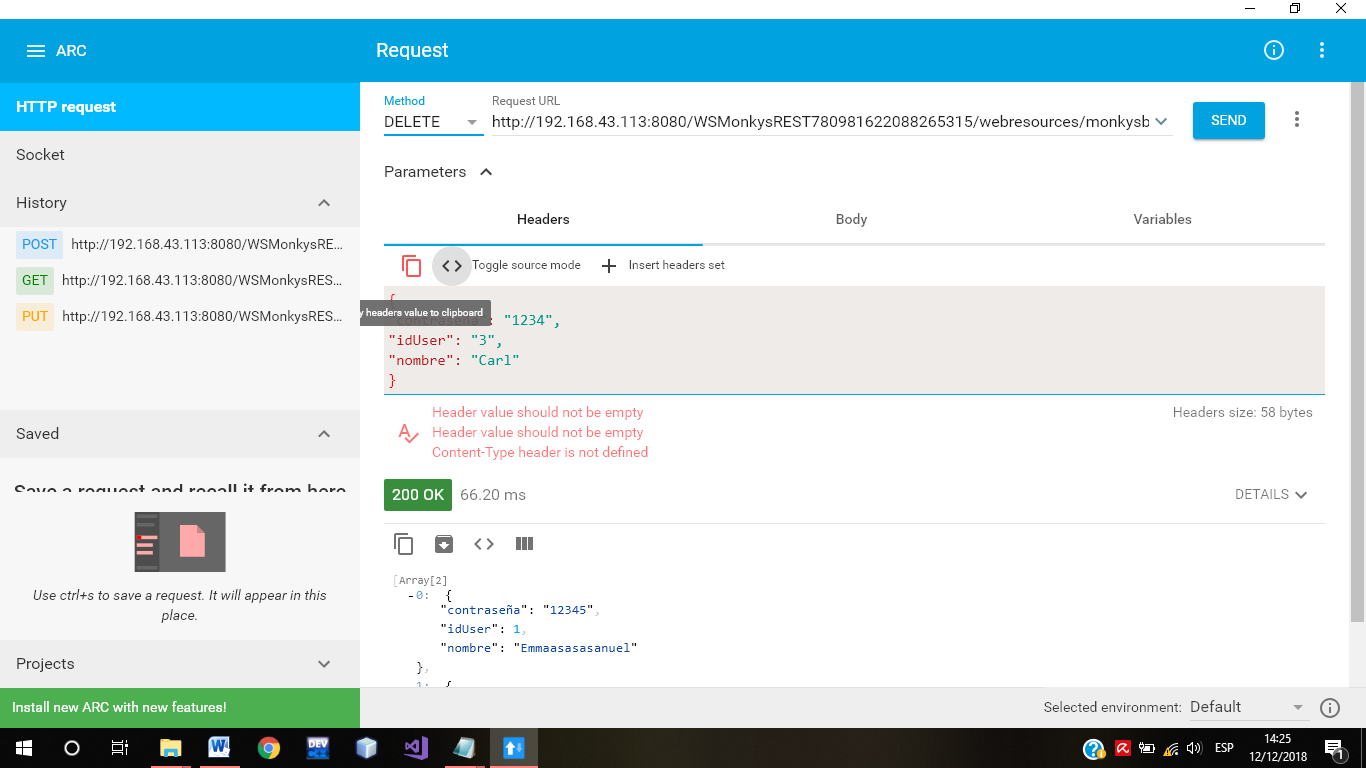
|  |  |  |  |
| --- | --- | --- | --- |
| BR# | Módulo | Roles utilizados | Descripción |
| B1 | Servicio web REST | **Usuario** | **Usuario:** puede ser capaz de ingresar, ver y eliminar datos de manera remota. |

**ScreenShot:**

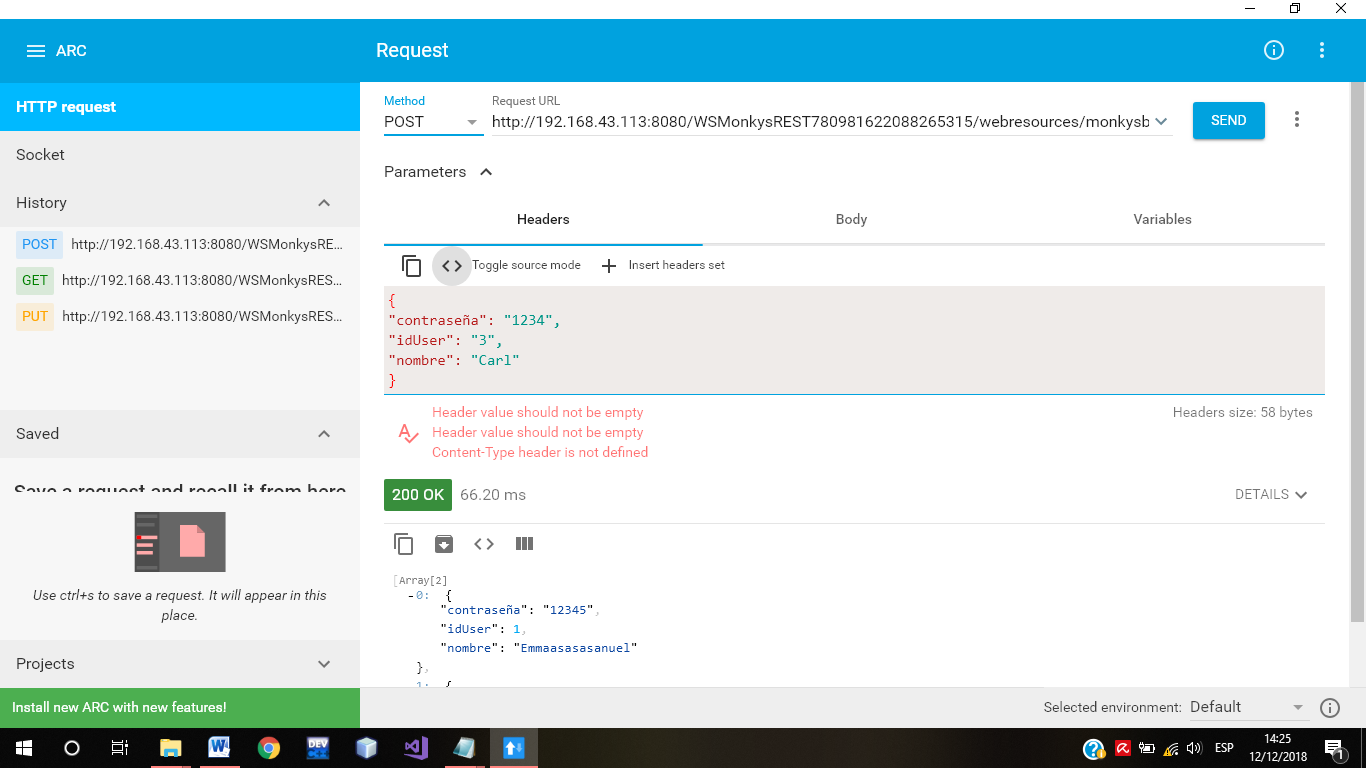
**T1.**

****

**T2. **

**T3. **

**T4.**

****

***Plan de Pruebas (Cliente).***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| BR | TR | Caso de prueba# | Caso de prueba | Pasos | Datos de Prueba | Resultado |
| B1 | T1 | 1 | Que no inicie sesion con usuarios inválidos | 1)Introducir un usuario  2) Presionar botón de inicio de sesión | Usuario: Pedro  Password:1234 | Mensaje de que no puedes iniciar sesión usuario y/o contraseña inválidos |
| B1 | T1 | 2 | Que no inicie sesión con campos vacíos. | 1)Presionar botón de iniciar sesión | Usuario: “ “  Password: “ “ | Mensaje de que no puedes iniciar sesión usuario y/o contraseña inválidos |
| B1 | T2 | 3 | Insertar | 1. Introducir el ID del Producto 2. Introducir el nombre 3. Introducir la cantidad 4. Introducir el precio de venta 5. Introducir una descripción del producto 6. Guardar | ID:8  Nombre: Cerveza  Cantidad: 100  Precio: $17  Descripción: “ “ | Mensaje de articulo guardado y se muestra en la tabla. |
| B1 | T2 | 4 | Modificar | 1. Introducir ID 2. Buscar 3. Introducir una descripción del producto 4. Guardar | ID:8  Descripción: “es una cerveza” | Mensaje de articulo modificado |
| B1 | T2 | 5 | Ver | 1. Introducir ID 2. Buscar | ID:8 | Muestra el articulo con el ID que se busca, si le dejamos el campo vacío muestra todos los registros |
| B1 | T2 | 6 | Eliminar | 1. Introducir ID 2. Eliminar | ID:8 | Mensaje de articulo eliminado |

**Requerimientos técnicos:**

**T1.** Ninguno de los campos debe de estar en blanco.

**T2.** Al presionar cualquier botón debe hacer la acción que le corresponde correctamente.

**Requerimientos de negocio:**

|  |  |  |  |
| --- | --- | --- | --- |
| BR# | Módulo | Roles utilizados | Descripción |
| B1 | Interfaz gráfica | **Usuario** | **Usuario:** puede ser capaz de iniciar sesión, insertar, modificar, ver y eliminar datos. |

***Evaluaciones:***

**Carlos Chavero:**

* **Autoevaluación:** 80
* ***Emmanuel:*** 75
* ***Ian:*** 70

**Emmanuel García:**

* **Autoevaluación:** 70
* ***Carlos:*** 85
* ***Ian:*** 70

**Ian Escalante:**

* **Autoevaluación:** 75
* ***Carlos:*** 85
* ***Emmanuel:*** 80