Contenido

[1.7 Interface 2](#_Toc196415205)

[1.7.1. Aplicabilidad a distintos tipos de interfaces 2](#_Toc196415206)

[1.7.2.1. Atributo de interfaz 3](#_Toc196415207)

[1.7.3. Diagramas 4](#_Toc196415208)

[1.7.3.1. DC2 Realización de pedidos 5](#_Toc196415209)

[1.7.3.2 Interacción del Usuario con el Realización de pedidos 6](#_Toc196415210)

## 1.7 Interface

Su propósito principal es establecer claramente cómo esa entidad se comunica con el exterior, especificando tanto los servicios que ofrece como los que requiere. Este punto de vista es fundamental cuando múltiples componentes deben interactuar correctamente y de forma controlada, ya sea durante el desarrollo o al integrar subsistemas. Este apartado aborda diversas preocupaciones de diseño. Una de ellas es garantizar que las interfaces estén bien especificadas antes de ser implementadas, lo cual permite detectar errores de integración con anticipación. Además, cumple una función documental: los detalles que se registran aquí pueden ser usados por diseñadores, desarrolladores, testers e incluso como parte de manuales para usuarios técnicos.

En cuanto al contenido, esta sección identifica qué entidades están involucradas en la interacción. Describe cuáles son los servicios proporcionados por el componente, y cuáles necesita de otros. Luego detalla los mecanismos de interacción, que pueden incluir llamadas directas a funciones, intercambio de mensajes, acceso a bases de datos o el uso de eventos. Un aspecto clave del Interface Viewpoint es la descripción precisa de atributos de la interfaz. También se debe definir el orden esperado de las interacciones si hay secuencias específicas que deben seguirse.

## 1.7.1. Aplicabilidad a distintos tipos de interfaces

El punto de vista de interfaz no se limita únicamente a interfaces de software tradicionales como APIs o servicios web. También se aplica a interfaces de usuario, interfaces de hardware, o cualquier otro medio por el cual una entidad del sistema interactúe con otra. Esto incluye, por ejemplo, la forma en que un módulo se comunica con la base de datos, cómo se expone una funcionalidad a través de una interfaz gráfica, o cómo se envían datos a otro sistema mediante archivos o mensajes. Esta flexibilidad permite que el Interface Viewpoint se use en una amplia variedad de contextos, adaptándose a sistemas distribuidos, aplicaciones móviles, embebidas, o incluso entornos híbridos donde conviven múltiples tecnologías.

**1.7.2. Elementos del diseño de interfaz**

Se centra en los elementos que deben estar presentes en una descripción adecuada de la interfaz de una entidad de diseño. Esta descripción no se limita a señalar que existe una conexión entre dos partes del sistema, sino que se adentra en cómo se produce esa interacción, qué se espera de cada lado, y en qué condiciones ocurre esa comunicación. Para comenzar, se requiere identificar con claridad cuál es la entidad de diseño a la que pertenece la interfaz. Esto implica especificar el módulo, componente o subsistema dentro del cual se encuentra.

Posteriormente, se debe detallar cómo esta entidad se comunica con otras, lo cual incluye tanto los servicios que ofrece como los servicios que espera recibir de otras partes del sistema. Es decir, no basta con decir que una interfaz "permite el registro de pedidos", sino que se debe definir qué funciones específicas están disponibles, qué entradas esperan, qué salidas generan, y cómo se maneja cualquier posible error durante esa interacción.

Un aspecto clave en esta sección es que la forma de interacción debe quedar bien especificada. Esto significa indicar si la comunicación entre componentes se realiza mediante llamadas a procedimientos, intercambio de mensajes, acceso a una base de datos compartida o alguna otra técnica.

## 1.7.2.1. Atributo de interfaz

El atributo de interfaz incluye tanto los mecanismos de interacción como las reglas que controlan esas interacciones. Entre los mecanismos se consideran formas de invocar, interrumpir o comunicarse con la entidad, ya sea por paso de parámetros, áreas de datos comunes, mensajes, o acceso directo a datos internos.

Las reglas de interacción abarcan el protocolo de comunicación, el formato de los datos, los valores permitidos y el significado de esos valores. También se describe el rango de entradas válidas, el formato y tipo de cada entrada o salida, y los posibles códigos de error.

En sistemas de información, este atributo debe incluir también los formatos de pantalla, los tipos de entrada, y si aplica, el lenguaje interactivo usado en la interfaz.

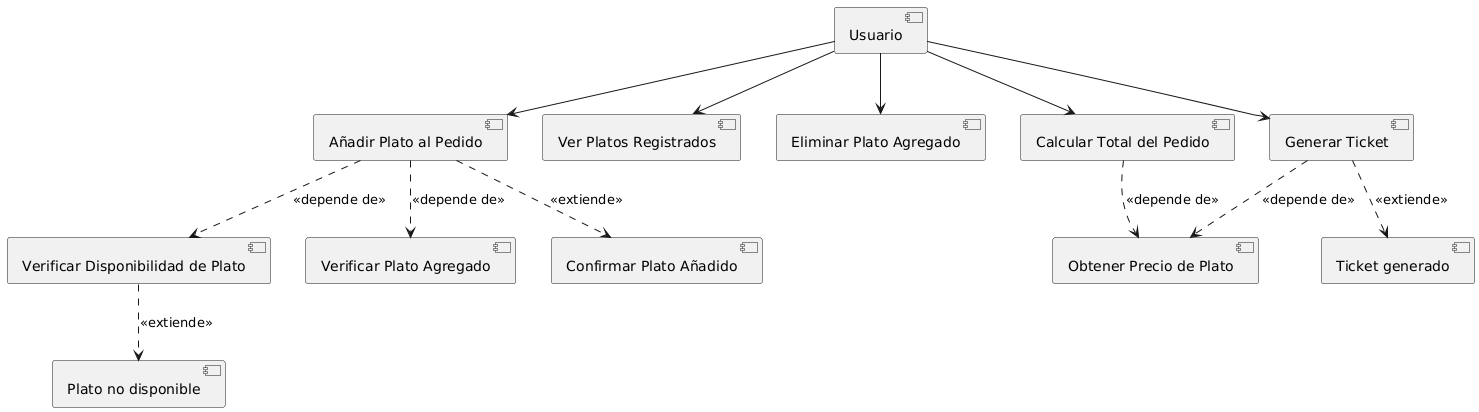
Este atributo es especialmente útil en proyectos sin interfaces externas, ya que permite documentar con precisión las interacciones entre componentes internos del sistema, como por ejemplo entre módulos de gestión de pedidos, usuarios y base de datos​

## 1.7.3. Diagramas

La sección 5.8.3 del estándar IEEE 1016 se enfoca en los lenguajes y representaciones utilizadas para describir las interfaces del sistema. En este punto se busca mostrar de forma clara cómo interactúan los usuarios con el software, detallando tanto las entradas permitidas como los resultados esperados a través de la interfaz. Para lograrlo, se hace uso de diagramas y representaciones gráficas que ilustran el funcionamiento visual y estructural de dichas interfaces. En el contexto del sistema de comanda digital, se incluyen diagramas que permiten visualizar la distribución de las pantallas, las opciones disponibles para cada tipo de usuario (como meseros, cocineros y administradores), así como los flujos de interacción que se generan durante el uso del sistema. Estas representaciones facilitan la comprensión de la interfaz y permiten verificar que su diseño cumple con los requisitos funcionales establecidos.

## 

## 1.7.3.1. DC2 Realización de pedidos



## 1.7.3.2 Interacción del Usuario con el Realización de pedidos

* **Usuario-> añadir plato al pedido**

El usuario comienza explorando los platos disponibles a través de la opción Ver Platos Registrados, donde el sistema le presenta un menú completo con nombres, descripciones y precios de cada opción. Una vez que el usuario selecciona un plato para agregar a su pedido, el sistema verifica su disponibilidad. Si el plato no está disponible, muestra un mensaje de error claro: El plato no está disponible". En caso contrario, confirma la acción con un mensaje de éxito: Plato agregado correctamente".

* **Usuario -> Ver Platos Registrados (VerPlatos)**

La interacción del Usuario con el componente "Ver Platos Registrados" es directa. Cuando el Usuario solicita ver los platos que el sistema tiene registrados, este componente accede a la información almacenada de los platos y la presenta al Usuario. En el diagrama actual, no se muestran dependencias directas con otros componentes para esta funcionalidad, lo que sugiere que "Ver Platos Registrados" se encarga de obtener y mostrar la información por sí mismo. La función principal de este componente es simplemente mostrar la lista de platos disponibles al Usuario para su consulta

* **Eliminar Plato Agregado (EliminarPlato)**

Cuando el Usuario decide eliminar un plato que ya había añadido a su pedido, interactúa con el componente "Eliminar Plato Agregado". El Usuario debe indicar qué plato desea remover de su pedido, y este componente se encarga de realizar la modificación correspondiente en la información del pedido del Usuario. Aunque no se muestran dependencias explícitas en el diagrama, se podría inferir que este componente interactúa con los datos del pedido actual del Usuario para llevar a cabo la eliminación. La acción principal de este componente es actualizar el pedido del Usuario, removiendo el plato seleccionado.

* **Usuario-> Calcular Total del Pedido**

Cuando el usuario solicita *Calcular Total del Pedido*, el sistema procesa los precios individuales de todos los platos agregados y muestra el monto total a pagar. Finalmente, al optar por *Generar Ticket*, el sistema produce un resumen detallado que incluye la lista completa de platos seleccionados, sus precios individuales, el total de la compra y la fecha y hora de emisión, ya sea en formato digital o impreso.

* **Usuario -> Calcular Total del Pedido (CalcularTotal)**

La interacción del Usuario con el componente "Calcular Total del Pedido" ocurre cuando el Usuario desea conocer el costo total de los platos que ha agregado a su pedido hasta ese momento. El Usuario podría accionar un botón o una función específica dentro de la interfaz del sistema para solicitar este cálculo.

* **Usuario -> Generar ticket**

La interacción del Usuario con el componente "Generar Ticket" se da cuando el Usuario ha finalizado su pedido y desea obtener un comprobante. Al igual que con el cálculo del total, el Usuario probablemente activará una opción específica en la interfaz para iniciar la generación del ticket. El diagrama indica que el componente "Generar Ticket" también *depende de* "Obtener Precio de Plato". Esto sugiere que, al igual que para calcular el total, el componente necesita acceder a los precios individuales de los platos en el pedido para poder incluirlos detalladamente en el ticket

* **Añadir Plato al Pedido-> Verificar Disponibilidad de Plato**

Cuando el componente **Añadir Plato al Pedido** recibe una solicitud interna del sistema (originada por la acción del Usuario de querer agregar un plato), su primer paso crucial es determinar si el plato en cuestión está actualmente disponible. Para lograr esto, el componente **Añadir Plato al Pedido** establece una comunicación con el componente **Verificar Disponibilidad de Plato**. Esta comunicación se realiza mediante una llamada o invocación de una función o servicio expuesto por el componente **Verificar Disponibilidad de Plato**.

* **Añadir Plato al Pedido-> Verificar Plato Agregado**

Una vez que el componente Añadir Plato al Pedido ha determinado que el plato está disponible, o incluso en paralelo a esta verificación, necesita considerar si el plato ya ha sido agregado previamente al pedido del Usuario. Para esto, interactúa con el componente Verificar Plato Agregado. El componente Añadir Plato al Pedido comunica al componente Verificar Plato Agregado la información del plato que se intenta añadir y el identificador del pedido del Usuario.

* **Interacción entre "Añadir Plato al Pedido" y "Confirmar Plato Añadido"**

La interacción entre el componente Añadir Plato al Pedido y Confirmar Plato Añadido ocurre una vez que el plato ha sido exitosamente añadido (o su cantidad actualizada) en el pedido del Usuario. Esta interacción representa una extensión de la funcionalidad principal de "Añadir Plato al Pedido". Cuando el componente Añadir Plato al Pedido completa la lógica para agregar el plato al pedido, activa o utiliza el componente Confirmar Plato Añadido. Esto podría implicar pasarle información sobre el plato que se añadió (nombre, cantidad, precio unitario, etc.) y el pedido al que se añadió.

* **Verificar Disponibilidad de plato-> plato no disponible**

El componente Verificar Disponibilidad de Plato tiene como responsabilidad principal determinar si un plato específico solicitado por el Usuario (a través del componente "Añadir Plato al Pedido") puede ser añadido a su pedido en ese momento. Para llevar a cabo esta verificación, este componente consulta diversas fuentes de información, como el inventario actual de ingredientes, la capacidad de la cocina, o cualquier otra regla de negocio que pueda afectar la disponibilidad de un plato

* **Calcular total del pedido -> obtener precio de plato**

Cuando el componente Calcular Total del Pedido necesita llevar a cabo su función, que es determinar el costo total de los artículos que el Usuario ha añadido a su pedido, requiere acceder a la información de los precios individuales de cada uno de esos artículos. Para obtener estos precios, el componente Calcular Total del Pedido establece una relación de dependencia con el componente Obtener Precio de Plato.

* **(Generar ticket) -> obtener precio de plato**

Cuando el Usuario finaliza su pedido y solicita la generación de un ticket o comprobante (a través de una acción en la interfaz que activa el componente Generar Ticket), este componente necesita acceder a diversos detalles del pedido para poder construir el ticket, y uno de los detalles cruciales es el precio de cada uno de los platos incluidos en el pedido. Para obtener esta información de precios, el componente Generar Ticket establece una relación de dependencia con el componente Obtener Precio de Plato.

* **Generar ticket -> ticket generado**

El componente Generar Ticket tiene como responsabilidad principal tomar la información del pedido final del Usuario, incluyendo los detalles de los platos, sus cantidades y los precios (obtenidos del componente Obtener Precio de Plato), y formatearla en un documento que sirva como comprobante o factura, es decir, el ticket. Una vez que el componente Generar Ticket ha completado el proceso de recopilación de la información necesaria, ha realizado los cálculos pertinentes (como subtotales, impuestos y el total final) y ha formateado esta información en la estructura del ticket, se produce la interacción con el componente Ticket generado.