

Comparador de tarifas:

Clases:

Tarifa: Almacena los datos de cada tarifa, una tarifa puede ser una tarifa de móvil o de fijo, que contiene datos adicionales a los de una tarifa común, por eso se relaciona con una generalización .

Atributos:

precio: Almacena el precio de la contratación de la tarifa.

url: Link a la página oficial de la tarifa.

permanencia: Cantidad de meses que se tiene de permanencia.

ofertas: Ofertas de cada tarifa. Puede ser una o muchas. Son ofertas: regalos de móvil, gb gratis a partir de cierto gasto, etc... Se guarda como una frase en un String.

puntuacionMedia: Puntuación media de la tarifa.

Métodos:

buscarTarifas: Recibe los filtros para las tarifas en base a una búsqueda del usuario.

aplicarFiltros: Aplica los filtros a la búsqueda.

MostrarTarifas: Devuelve las tarifas encontradas en la búsqueda

obtenerComentarios: Pide a una tarifa todos los comentarios.

MostrarComentarios: Muestra por pantalla los comentarios obtenidos de la tarifa.

AccederWebCompañia: Redirecciona al usuario a la web de inicio de la compañía.

TarifaMovil: Contiene tarifas de móvil. Es una tarifa con datos adicionales, por eso es una generalización de Tarifa.

Atributos:

gb: Cantidad de datos de Internet a contratar.

precioLlamadas: Precio de las llamadas para tarifas de móvil.

precioSMS: Precio de los SMS para tarifas móvil.

TarifaFijo: Contiene tarifas de teléfono fijo y si incluye ADSL o Fibra Óptica. Es una tarifa con datos adicionales, por eso es una generalización de Tarifa.

Atributos:

Establecimiento: Precio de establecimiento de llamada.

MinutosAFijo: Cantidad de minutos que se puede hablar de forma gratuita con números fijos.

MinutosAMovil: Cantidad de minutos que se puede hablar de forma gratuita con números móviles.

ADSL: ADSL ofertado.

Atributos:

mb: Velocidad ofertada por la tarifa de ADSL.

Fibra: Fibra Óptica ofertada.

Atributos:

mb: Velocidad ofertada por la tarifa de Fibra.

Simetrica: Booleano que indica si es simétrica o no.

Compañía: Base de datos con todas las compañías de las que se van a extraer las tarifas. Es la clase principal del sistema ya que de esta dependen las tarifas que se van a filtrar.

Atributos:

nombre: Nombre de la compañía

pais: País donde opera la compañía

web: Url de la página de inicio de la compañía

Comentario: Opinión del cliente sobre la tarifa o compañía. Esto engloba a su puntuación dada y su redacción opinando sobre lo puntuado. No es posible puntuar sin comentar y viceversa.

Atributos:

puntuacion: Puntuación dada por el usuario a la tarifa o compañía.

opinion: Texto escrito que el usuario publica como comentario. La opinión puede ser sobre la tarifa o sobre la compañía que la oferta, por ello no se ofrece la posibilidad de comentar cada compañía en un apartado independiente.

Métodos:

getComentarios: Obtiene el comentario (uno solo) para pasarlo en bucle a la tarifa y obtenerlos así todos.

PublicarComentario: Publica el comentario solicitado previa confirmación.

Usuario: Almacena los datos de usuario.

Atributos:

nombre: Nombre de usuario.

email: Correo electrónico de registro.

contraseña: Contraseña del usuario.

Métodos:

iniciarSesion: Método para iniciar sesión de usuario.

registrarse: Registro de usuario.

ComprobarSesion: Comprueba si la sesión está iniciada para permitir al usuario comentar o llevarle a la página de inicio de sesión si no.

Administrador: Usuario con privilegios adicionales sobre la web. Puede modificar compañías y tarifas, borrar comentarios y tarifas y banear usuarios.

Métodos:

modificarCompañía: El Administrador modifica los parámetros de una compañía o la elimina.

ModificarTarifa: El Administrador modifica los parámetros de una tarifa o la elimina.

BorrarComentario: El Administrador borra los comentarios que considera oportuno. El Administrador no puede modificar un comentario de un usuario, solo eliminarlo.

banearUsuario: El Administrador puede eliminar un usuario de la base de datos para evitar que vuelva a iniciar sesión. Al hacer esto se borran sus comentarios en todas las tarifas.

Consultor: Subsistema que accede a la API para obtener las tarifas y actualizar la base de datos del comparador (sistema general principal).

Métodos:

actualizarBD: Revisa las tarifas obtenidas de la API y almacena en la base de datos del comparador las tarifas nuevas o actualiza las desactualizadas.

actualizarTarifas: Se comunica con la API y obtiene todas las tarifas de cada compañía.

API: Será la API de cada compañía con la que se comunicará el subsistema consultor para realizar consultas.

Relaciones:

Compañía - Tarifa: Relación de composición. Una compañía tiene 0 o muchas tarifas. Si la compañía es nueva o el administrador la incluye a mano puede no tener tarifas aún. Tarifa depende de compañía ya que una tarifa esta dentro de una compañía.

Tarifa - TarifaMóvil: Relación de generalización. Una tarifa móvil es un tipo de tarifa.

Administrador - Usuario: Relación de generalización. Un Administrador es un Usuario con ciertos permisos extra.

Comentario – Usuario: Relación entre los usuarios y los comentarios que han publicado.

(Comentario, Usuario) - Tarifa: Relación de clase de asociación. Comentario y Usuario se relacionan a través de una tarifa, donde el usuario publica un comentario.

Diagramas de secuencia:

Cliente: Usuario con privilegios estándar (no administrador) que entra al sistema para hacer consultas o publicaciones.

Un usuario entra al sistema y tiene la opción de buscar tarifas, ver comentarios, registrarse, iniciar sesión o publicar comentarios.

El usuario busca tarifas: El usuario selecciona una serie de filtros que se pasan al buscador de tarifas con el método de Tarifa buscarTarifas(). Estos filtros se pasarían a la clase Tarifa, que contiene el método aplicarFiltros(), que recibe los filtros de la búsqueda y los aplica, obteniendo así las tarifas encontradas, y devolviéndolas al usuario en forma de página de búsqueda con las tarifas encontradas con el método mostrarTarifas(), que es un método de Tarifa.

El usuario ve comentarios: El usuario decide ver comentarios y la clase Tarifa realiza la acción con el método verComentarios(). La clase Tarifa llama a la clase Comentario, que mediante el método getComentarios() devuelve un comentario a la clase Tarifa, y esta a su vez devuelve el comentario con el método mostrarComentarios().

El usuario se registra: El cliente quiere registrarse, y para ello el sistema llama a la clase Usuario con el método registrarse(), que almacena al usuario y sus credenciales en la base de datos del sistema como un nuevo usuario de clase Usuario.

El usuario inicia sesión: El cliente quiere iniciar sesión, y para ello el sistema llama a la clase Usuario con el método iniciarSesion(), que comprueba si el usuario está registrado e inicia sesión en consecuencia a ello o le remite a la página de registro.

El usuario publica un comentario: El usuario quiere publicar un

comentario en una tarifa, y para ello se llama al método `almacenarComentario()`, que recoge el comentario escrito por el usuario y su puntuación dada y comprueba si el usuario ha iniciado sesión llamando a la clase usuario con el método `comprobarSesión()`. La clase usuario ejecuta este método y devuelve el estado de la sesión, que en caso de no estar activa da al cliente el mensaje de `sesionNoIniciada()`, que redirecciona al cliente a la página de inicio de sesión. Si la sesión estaba correctamente iniciada, el sistema publica el comentario recogido previamente con el método `publicarComentario()` de la clase `Comentario`, que almacena este comentario en la base de datos.

El usuario accede a la web: El usuario quiere ir a la página web de la compañía y para ello el sistema llama a la clase `Tarifa` con el método `accederWebCompañía()`.

Administrador: Usuario con privilegios elevados que entra al sistema para hacer cambios, eliminar contenido o actualizar información.

El administrador del sistema accede para realizar los cambios.

El administrador puede modificar una compañía: Llamando al método `modificarCompañía()`, que aplica los cambios efectuados por el administrador a la compañía seleccionada. No devuelve nada al administrador.

El administrador puede bloquear un usuario: Llamando al método `bloquearUsuario()`, que elimina al usuario de la base de datos. No devuelve nada al administrador.

El administrador puede eliminar un comentario: Llamando al

método eliminarComentario(), que borra el comentario de la compañía seleccionada. No devuelve nada al administrador.

El administrador puede modificar una tarifa: Por si hay algún dato erróneo llamando al método modificarTarifa(), que aplica los cambios efectuados por el administrador a la tarifa seleccionada. No devuelve nada al administrador.

Consultor y API: El consultor es el subsistema que se encarga de pedir a la API los datos de las tarifas y compañías.

Para ello el consultor contiene el método actualizarTarifas() que pide a la API las tarifas actualizadas.

Tras esto la API devuelve lo que se le pidió, y con el método actualizarBD() se registran las modificaciones y actualizan tarifas o añaden tarifas que no existían a la base de datos del comparador.

Diagrama Casos de uso:

Usuario: Puede ver tarifas o iniciar sesión.

Si decide ver tarifas, el sistema le da la opción desde la página de búsqueda de hacer uso de los filtros para ver las tarifas disponibles, ver comentarios de las tarifas mostradas o ir a la página web de una de las tarifas.

Si el usuario quiere iniciar sesión debe registrarse antes.

Para acceder a la opción de comentar, el usuario debe siempre tener una sesión iniciada previamente.

Administrador: Para acceder a privilegios de administrador, este debe siempre iniciar sesión antes, y tras esto puede Administrar,

que consiste en Modificar Tarifas, Eliminar Comentario, Modificar Compañías y Bloquear Usuario.

Consultor y API: El consultor (que es un subsistema que se encarga de manejar la base de datos principal del comparador de tarifas) obtiene datos de la API regularmente para actualizar la base de datos del comparador de tarifas.

“Comparador de tarifas” es el nombre del sistema completo.