

Informe sobre integralces

fecha: 12 de julio 2015

Indice

- [2303067-Bad user name in account debited message](#)
- [2303059-The "Last trade" box shows the first one](#)
- [2303063-Expire date in offer edit form](#)
- [2303155-Sort accounts by balance](#)
- [2303659-Edit blog post form doesn't have title](#)
- [2303669-The "Account statement" page has incorrect](#)
- [2300359-Add Delete users account](#)
- **[2308405-Unable to delete limit from limit chain](#)**
 - [Documentación interna en 2308405-Unable to delete limit from limit chain](#)
- [Menú Account users](#)
- [Lista de usuarios con apellidos](#)
- [Total general](#)

2303067-Bad user name in account debited message

Las "cuentas debitado" mensajes muestra el nombre del vendedor en lugar del nombre del vendedor en la cabecera del mensaje (Estimado @ nombre de usuario).

2303059-The "Last trade" box shows the first one

The "Last trade" box in summary block shows the first trade instead of the last one.

2303063-Expire date in offer edit form

El campo "fecha de caducidad" en el formulario de edición oferta no funciona bien. Se pone a cero al 12/1/1999 al editar una oferta. No sé si lo mismo ocurre con los deseos.

2303155-Sort accounts by balance

Cuando la tabla de pedido cuentas por el equilibrio en el CES página / admin / explicar las filas quedan ordenados alfabéticamente por el equilibrio en lugar de numéricamente, por lo que por ejemplo tenemos

1.20 CUR 13.5 CUR 2.0 CUR

pero debemos tener

1.20 CUR 2.0 CUR 13.5 CUR

Tenga cuidado también con los negativos!

2303659-Edit blog post form doesn't have title

Edit blog post form doesn't have title

node/add/ces-blog

2303669-The "Account statement" page has incorrect

El texto de ayuda en la página Estado de cuenta es "Haga clic en el cambio que desea unirse." lo cual es incorrecto. Este texto de ayuda debe ser para la página antes de formulario de registro. La url de esta página es ces / banco / cuenta / transacción

Además, cuando no hay resultados Debemos mostrar un texto que indica que no hay ninguna operación en el intervalo de tiempo seleccionado y que se puede tratar con un intervalo más amplio. Ahora estamos mostrando una mesa vacía.

2300359-Add Delete users account

<https://www.drupal.org/node/2300359>

Esta función es compatible con el núcleo del CIEM y ya existe una interfaz para operar cuando un usuario ya tiene varias cuentas (menú desplegable en el bloque de síntesis).

Pero no hay una interfaz para crear estas cuentas adicionales o para compartir una cuenta entre los diferentes usuarios

Por otra parte, la importación de datos desde el CES puede crear muchos usuarios con el mismo email, y esto es un paso en el fin de solucionar problemas que.

Características solicitadas:

- 1 - Un administrador crea una cuenta de un usuario drupal existente (Está realizado pero puede ser que la interfaz actual sea obsoleta).
- 2 - Un usuario de Drupal existente solicita una cuenta adicional en su red de intercambios o para otras redes.
- 3 - Un administrador o un propietario de la cuenta Agrega o quita otros usuarios.

Tenga en cuenta que la ces_account_user tabla tiene un tercer campo, "privilegio", que es actualmente no se utiliza, pero que podría ser utilizado para diferenciar el propietario de la cuenta y los otros usuarios de la cuenta. La diferencia entre estos dos papeles podría ser sólo la posibilidad de añadir / eliminar usuarios de su / sus cuentas.

Tenga en cuenta también que la mayoría de los usuarios no van a usar esta función, por lo que la interfaz de usuario debe ser lo menos intrusiva posible.

2308405-Unable to delete limit from limit chain

Cómo reproducir el error:

1. Entrar con la cuenta de usuario administrador del intercambio
2. Ir a ces / admin / límite / nuevo (o por menú, Administración-> Límites-> Nuevo)
3. Crear una cadena de límite
4. Añadir un "límite de débito por defecto" con los parámetros (por ejemplo: bloque, -50)
5. Haga clic en "Actualizar Límite Chain" (el límite se ha añadido)
6. Haga clic en "Eliminar". (El límite se elimina de la lista).
7. Haga clic en "Actualización de la cadena de límite" Y el límite eliminado vuelve a aparecer en la lista límite!

Documentación interna en 2308405-Unable to delete limit from limit chain

docs/es/readme.dox:

@file

@brief Documentación para desarrolladores

@mainpage

Documentación para desarrolladores

Bienvenido o bienvenida a la documentación para desarrolladores del proyecto IntegralCES. Ésta es una documentación permanentemente en construcción que vamos completando a medida que nos es necesario.

@section about Acerca de

Integral CES es un conjunto de módulos de Drupal para la gestión de monedas sociales y sus comunidades.

ver [integralces.net](http://www.integralces.net) <<http://www.integralces.net>> para obtener más documentación y demo.integralces.net <<http://demo.integralces.net>> para un sitio de demostración.

@section alta_exchange Dar de alta nuevo exchange

Visitar el link 'New exchange'. Rellenar el formulario múltiple y enviarlo. El administrador lo activará, con lo que daremos de alta el grupo de intercambio y el usuario drupal que necesitas, con una cuenta vinculada al grupo de intercambio.

A partir de que demos de alta el exchange y tu usuario administrador, los otros usuarios de la eco-xarxa se tendrán que dar de alta usando el selector de la home (New to IntegralCES?), seleccionando el país, i pidiendo una cuenta en el exchange que acabamos de dar de alta. Esas peticiones de usuarios ya tendrían que llegar al administrador, Se activan y listo!

@section modulo_raiz Módulo raíz CES

El módulo de la raíz se denomina ICES. Este módulo por sí solo no proporciona ninguna función para el usuario final. Tiene un par de objetivos:

- Se trata de un contenedor para todos los demás módulos en una carpeta principal por lo que facilitar la instalación y actualización.
- Proporcionar algunas bibliotecas compartidas para ser utilizados por cualquiera de los submódulos.

Los archivos que pertenecen directamente al módulo de raíz son incluyen la `ces.info` obligatorio y `ces.module` y las bibliotecas compartidas. Actualmente hay una biblioteca compartida. Consiste en un capa de abstracción de base de datos ubicada en el archivo `commons.db.logic.inc`.

Base de datos

Hay una forma común para acceder a la base de datos de Drupal en todos los sub-módulos CES. Se trata de un versión aproximada del [patrón de registro activo](http://en.wikipedia.org/wiki/Active_record_pattern). El objetivo de esta capa es concentrar toda la el trabajo de base de datos en un solo lugar estandarizada. Aísla a otro código de la capa de base de datos de Drupal y aquellos que facilita la detección de errores, permite un sistema de captura

centralizada y hace que la adaptación a API Drupal cambia casi trivial. Esta interfaz común se define en el archivo commons.db.logic.inc.

Este archivo define varias clases:

- IcesDBObject: La clase abstracta que cualquier objeto persistente debe extenderse. Por ejemplo, la clase Exchange extiende IcesDBObject. Para cada clase se extiende DBOBJECT hay una tabla en la base de datos. Esta clase tiene una propiedad crucial llamada table. Una clase hija debe establecer esta variable para el nombre de tabla asociado con él. También tienen implementaciones por defecto para los dos métodos que convierten las variables de objeto interno en un array asociativo se salven y viceversa. Por objeto predeterminado propiedades públicas se convertirán en campos de la tabla.
- IcesSerializer: Esta clase tiene métodos para guardar, actualizar, cargar y eliminar cualquier objeto de tipo IcesDBObject. Es el puente real con la capa de base de datos de Drupal. Más allá de la interface CRUD básica, que ofrece algunos otros métodos de ayuda que también cuenta con una serialización especial estrategia para las jerarquías de clases que implementan el [patrón decorador](https://es.wikipedia.org/wiki/Decorator_%28patr%C3%B3n_de_dise%C3%B1o%29).
- IcesLockSerializer: Esta es una subclase de IcesSerializer para aquellas operaciones que son delicadas debido a la concurrencia. Esta clase asegura que cuando se carga un objeto, que no será cargado por cualquier otro LockSerializer hasta que lo guarde.
- DbTransaction: Un contenedor vacío para el módulo CesBank transaction.

![Base de datos](../img/database.png)

@section modulo_ces_bank Módulo CES BANK

Este es el módulo bancario principal. Define y trabaja con todos los objetos de banca: grupos de intercambio (exchnages a partir de ahora), cuentas, transacciones ... Este módulo se divide en dos partes:

- La API de banco (@subpage ces_bank.logic.inc) es el responsable de hacer toda la lógica de la banca. Se encuentra en el fichero bank.logic.inc. Cuenta con una interfaz pública para crear cuentas, hacer transacciones, etc
- El módulo de banco (@subpage ces_bank) es propiamente el módulo de Drupal con su .info y archivos module y otros archivos de páginas y formularios. Utiliza la API de banco para realizar las acciones ordenadas por la interfaz de usuario web.

La API del banco

La API del banco contiene toda la lógica de la gestión de cuentas. Se organiza en un conjunto de clases. Hay sólo una clase pública que actúa al mismo tiempo como un controlador para el conjunto y como interfaz publica. Todas las otras clases son privadas y no deben utilizarse desde fuera. Incluso los parámetros públicos de la clase Bank no están accesibles para los otros los tipos de clases. La API banco debe ser pensada como una biblioteca externa. En particular, es independiente de Drupal. Por la forma en que podría ser utilizado en integraciones dentro de otras plataformas. Utiliza las funciones de acceso a base de datos desde el módulo raíz CES y utiliza las funciones de

mensajería del módulo @subpage ces_message. A continuación se comenta el director clases y sus atribuciones.

- **CesBank**: Interfaz pública para todas las funciones previstas para esta API. Los parámetros y el regreso Los valores de esta clase son, cuando se necesitan datos estructurados, arrays asociativos. El trabajo es diferido a los objetos particulares concretos. Una implementación típica función de esta clase es recibir y (mínimamente) analizar la entrada, cree el objeto privado apropiado, llamar a este objeto de función adecuada, (mínimamente) analizar el resultado, y devolverlo a la persona que llama.
- **CesBankExchange**: Representa un grupo de intercambio de comercio. Tiene un nombre, un código de 4 letras, y la cúpula opciones configurables.
- **CesBankAccount**: Es una interfaz que representa una cuenta bancaria. Actualmente sólo hay una implementación de esta interfaz. Se espera para el futuro utilizar esta interfaz para recoger cuentas a distancia.
- **CesBankLocalAccount**: Implementación de la cuenta para las cuentas almacenadas en la base de datos local. Una cuenta pertenece a un intercambio particular. Cada uno tiene un identificador único, comenzado con el código de 4 letras de su intercambio, seguido de 4 números. Especial cuentas pueden terminar con 4 letras en lugar de números. Cada LocalAccount es potencialmente relacionadas con uno o más usuarios, y cada usuario puede tener uno o más cuentas de este tipo. Cada cuenta tiene también un LimitChain, que determina la límites para esta cuenta especial de crédito y débito (y otros).
- **CesBankAccountLimit**: Límite abstracto Atómica para una cuenta. Es la piedra angular de LimitChains.
 - **AbsoluteDebitLimit**: Limita el débito total de una cuenta.
 - **AbsoluteCreditLimit**: Limita el crédito total de la cuenta.
- **CesBankLimitChain**: Un conjunto ordenado de AccountLimit, identificado con un nombre. Cada cuenta tiene un enlazar con un objeto de este tipo. Una cuenta satisface una LimitChain si satisface todas sus CesBankAccountLimits. Un intercambio puede configurar varios CesBankLimitChains para varios tipos de cuentas. Por ejemplo, un 'default' para los nuevos miembros, una "organización" para las cuentas compartidas que necesitan más flexibilidad y una "pública" para las cuentas públicas sin ningún tipo de límite.
- **CesBankTransactionInterface**: Interfaz para la transacción entre dos cuentas. Tiene métodos para acceso a ambos códigos de cuenta, la cantidad y la descripción o el concepto de la transacción. Existen varias implementaciones utilizando un esquema de patrón decorador.
 - **CesBankBasicTransaction**: Transacción entre dos cuentas del mismo intercambio.
 - **CesBankDecoratedTransaction**: Resumen de transacciones que delega todos los métodos para una transacción "padre" que contiene como una propiedad. Esta clase puede heredar añadir funciones a una transacción.
 - **CesBankInterExchangeTransaction**: Hijo de DecoratedTransaction para permitir el comercio entre cuentas de distintos intercambios.

ciclo de vida de una transacción.

Las transacciones tienen sólo dos operaciones: aplicar una nueva transacción y revocar una aplicada anteriormente. Sin embargo, una transacción pasará por varios estados entre que se crea y pasa a ser efectiva e irreversible.

Este ciclo de vida se resume en el siguiente diagrama que ahora será comentado estado por estado.

- NEW: Estado inicial. La transacción ha sido creada pero aún no se ha activado. Este es el estado mientras la transacción puede ser editada. A la espera de que el usuario confirme la transacción.
- TRIGGERED: La transacción está a la espera de ser aceptada. Inmediatamente después de que se pide aceptación por parte de todas las autoridades pertinentes. Se está a la espera de su respuesta. Se pasará a ACCEPTED o REJECTED.
- ACCEPTED: La transacción ha sido aceptada por todas las autoridades. Este estado tiene una vida muy corta en las transacciones locales. El siguiente estado es ERROR si el sistema no aplica la transacción, por ejemplo, un servidor remoto no pudo responder, o COMMITTED.
- COMMITTED: La transacción se aplicó con éxito. Sin embargo, puede ser revocada. Si hay no son temas para esta transacción después de un tiempo prudencial, que se archivará.
- ARCHIVED: La transacción está archivada y aplicada con éxito. Nada se puede hacer con ella. Existe sólo como datos históricos.
- REJECTED: La transacción, previamente disparada para ser aplicada, ha sido rechazada por algunos de las autoridades. Se desecha después de un tiempo prudencial. Puede pasar a ACCEPTED si algunas autoridades rectifican.
- DISCARDED: La transacción se descarta y no se ha aplicado. Nada se puede hacer con ella. Existe sólo como datos históricos.
- ERROR: Se produjo un error del sistema al aplicar o revocar la transacción. El administrador debe tener cuidado de él.

Estos fueron los distintos estados que una transacción puede pasar, A continuación los estados relacionados con la operación de revocación. El estado inicial de la operación de revocación es el de COMMITTED.

- REVOKE TRIGGERED: La transacción se ha disparado para ser revocada. Esperando la aceptación de la operación por parte de todas las autoridades.
- REVOKE ACCEPTED: La operación de revocación ha sido aceptada por todas las autoridades. El sistema está revocando efectivamente la transacción. Este estado es muy efímero en transacciones locales. El siguiente estado será REVOKED o ERROR si el sistema no revoca la transacción.
- REVOKE REJECTED: Algunas autoridades no aceptaron la operación de revocación. La transacción seguirá en estado COMMITTED y pasará a

ARCHIVED después de un tiempo prudencial.

- REVOKED: La transacción ha sido revocado con éxito. Pasara a DISCARDED después de un tiempo prudencial.

El estado de la transacción se cambia el uso de la función `setstate()` de la interfaz de transacción. Este función también desencadena las acciones apropiadas. Por ejemplo, cuando el estado se cambia de TRIGGERED a ACCEPTED, que inicia la aplicación efectiva de la transacción.

![Estados de las transacciones](../img/states_transaction.png)

CES Banco diagrama de clases del módulo

Diagrama de clases para el módulo bancario. Es una aproximación a la implementación real: hay muchos más propiedades y métodos, y algunos de ellos pueden tener nombres distintos. Sin embargo, la esencia arquitectónica es correcta.

![Diagrama de clases](../img/class_diagram.png)

Algunos componentes de IntegralCES

- @subpage install
- @subpage flujo_trabajo
- @subpage ces_permissions_doc
- @subpage external_transactions_doc

Menú Account users

Hola!

Volia proposar de treure el link del menú de l'esquerra i posar-lo més amagat, ja que és una feature que no esperem que sigui massa utilitzada i així mantenim el menú de l'esquerra ben net.

Per exemple podríem posar un link al veure el compte d'un usuari, a l'apartat del compte corrent, que digués

"Other account users: Maria, José, ..."

Que si tens permisos sigui un link i que al clicar et porti a la pàgina on et porta ara el menú de l'esquerra. Crec que no ha de ser difícil.

Lista de usuarios con apellidos

En el cambio de interfaz de la lista de usuarios ahora el nombre no aparece completo, sino sólo los apellidos.

Puedes utilizar una función del módulo `ces_user` para recuperar el nombre entero, o mira cómo se hacía antes. De memoria creo que algo así como `ces_user_get_fullname()` para generar el nombre completo.

Habría que arreglarlo pues ahora cuesta un poco de ver quién son los usuarios de cada cuenta.

ces/bank/exchange/accounts

Añadimos apellidos a la lista.

Total general

Totales	
Tiempo total (H:M)	81:05
Euros / hora	0€
Total global	0.00€