app/src/androidTest/java/org/gnucash/android/test/ui/GnucashAndroidTestRunner\_java,2

- Log.i ( TAG , " All animations disabled. " ) ;

+ Log.i ( TAG , " All animations enabled. " ) ;

- Log.i ( TAG , " Could not disable animations. " ) ;

+ Log.i ( TAG , " Could not enable animations. " ) ;

app/src/main/res/values-da-rDK/strings\_xml,10

- < string name = " wizard\_option\_currency\_other " > Other... < /string >

+ < string name = " wizard\_option\_currency\_other " > Other < /string >

app/src/main/res/layout/fragment\_line\_chart\_xml,1

- android:minHeight = " 400dp "

+ android:minHeight = " 300dp "

app/src/main/java/org/gnucash/android/ui/export/ScheduledExportListFragment\_java,9

- switch ( position ) {

+ switch ( position ) {

app/src/main/java/org/gnucash/android/ui/export/ScheduledExportListFragment\_java,10

- String dropboxAppKey = getString ( R.string.dropbox\_app\_key , SettingsActivity.DROPBOX\_APP\_KEY ) ;

- String dropboxAppSecret = getString ( R.string.dropbox\_app\_secret , SettingsActivity.DROPBOX\_APP\_SECRET ) ;

+ String dropboxAppKey = getString ( R.string.dropbox\_app\_key , SettingsActivity.DROPBOX\_APP\_KEY ) ;

+ String dropboxAppSecret = getString ( R.string.dropbox\_app\_secret , SettingsActivity.DROPBOX\_APP\_SECRET ) ;

- if ( ! mDbxAccountManager.hasLinkedAccount () ) {

+ if ( ! mDbxAccountManager.hasLinkedAccount () ) {

app/src/main/java/org/gnucash/android/ui/export/ScheduledExportListFragment\_java,5

- private static final String TAG = " ExportDialogFragment " ;

+ private static final String TAG = " ExportFormFragment " ;

app/src/main/java/org/gnucash/android/ui/settings/OwnCloudDialogFragment\_java,2

- private void checkdata () {

+ private void checkData () {

GnucashMobile/res/menu/global\_actions\_xml,1

- android:title = " @string/settings "

+ android:title = " @string/title\_settings "

app/src/main/java/org/gnucash/android/importer/CommoditiesXmlHandler\_java,1

- import org.gnucash.android.db.CommoditiesDbAdapter ;

+ import org.gnucash.android.db.adapter.CommoditiesDbAdapter ;

app/src/org/gnucash/android/db/DatabaseAdapter\_java,2

- public Cursor fetchRecord ( String tableName , long rowId ) {

+ protected Cursor fetchRecord ( String tableName , long rowId ) {

app/src/main/java/org/gnucash/android/db/adapter/BudgetAmountsDbAdapter\_java,5

- Split split = new Split ( balance.absolute () , accountUID ) ;

+ Split split = new Split ( balance.abs () , accountUID ) ;

app/src/test/java/org/gnucash/android/test/unit/db/TransactionsDbAdapterTest\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

- import org.gnucash.android.db.SplitsDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.SplitsDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

GnucashMobile/res/values-de/strings\_xml,1

- < resources > < string name = " app\_name " > Gnucash < /string >

+ < resources > < string name = " app\_name " > GnuCash < /string >

app/src/main/java/org/gnucash/android/ui/homescreen/WidgetConfigurationActivity\_java,1

- public static void updateWidget ( Context context , int appWidgetId , String accountUID ) {

+ public static void updateWidget ( final Context context , int appWidgetId , String accountUID ) {

- Account account ;

+ final Account account ;

app/src/main/java/org/gnucash/android/ui/homescreen/WidgetConfigurationActivity\_java,2

- RemoteViews views = new RemoteViews ( context.getPackageName () ,

+ final RemoteViews views = new RemoteViews ( context.getPackageName () ,

- Money accountBalance = accountsDbAdapter.getAccountBalance ( accountUID , - 1 , System.currentTimeMillis () ) ;

- views.setTextViewText ( R.id.transactions\_summary ,

+ Money accountBalance = accountsDbAdapter.getAccountBalance ( accountUID , - 1 , System.currentTimeMillis () ) ;

+

+ views.setTextViewText ( R.id.transactions\_summary ,

-

app/res/values-fr/strings\_xml,1

- < string name = " description\_add\_transaction\_icon " > Ajouter une nouvelle transaction &#224 ; un compte < /string >

- < string name = " label\_no\_accounts " > Aucun compte &#224 ; afficher < /string >

+ < string name = " description\_add\_transaction\_icon " > Ajoute une nouvelle transaction &#224 ; un compte < /string >

+ < string name = " label\_no\_accounts " > Aucun comptes &#224 ; afficher < /string >

- < string name = " label\_no\_transactions\_to\_display " > Aucune transaction &#224 ; afficher < /string >

+ < string name = " label\_no\_transactions\_to\_display " > Aucune transactions &#224 ; afficher < /string >

app/res/values-fr/strings\_xml,3

- < string name = " summary\_default\_currency " > Monnaie par d&#233 ; faut &#224 ; assigner aux nouveau comptes < /string >

+ < string name = " summary\_default\_currency " > Monnaie par d&#233 ; faut &#224 ; assigne aux nouveau comptes < /string >

app/res/values-fr/strings\_xml,2

- < string name = " title\_confirm\_delete " > Confirmation de suppression < /string >

- < string name = " delete\_account\_confirmation\_message " > Toutes les transactions dans ce compte seront supprimer < /string >

+ < string name = " title\_confirm\_delete " > Confirmer la suppression < /string >

+ < string name = " delete\_account\_confirmation\_message " > Toutes les transactions dans ce compte seront aussi supprimes < /string >

- < string name = " title\_selected " > %1$d s&#233 ; lectionner < /string >

+ < string name = " title\_selected " > %1$d s&#233 ; lectionn ( s ) < /string >

- < string name = " hint\_export\_choice " > Par d&#233 ; faut , seul les nouvelles transactions depuis le dernier export seront exporter. Cochez cette option pour exporter toutes les transactions < /string >

- < string name = " error\_exporting " > Erreur lors de l\'export des donn&#233 ; es en OFX < /string >

+ < string name = " hint\_export\_choice " > Par d&#233 ; faut , seul les nouvelles transactions depuis le dernier export seront exportes. Cochez cette option pour exporter toutes les transactions < /string >

+ < string name = " error\_exporting " > Erreur lors de l\'export des donn&#233 ; es en OFX < /string >

- < string name = " hint\_delete\_after\_export " > Toutes les transactions export&#233 ; seront supprimer apr&#233 ; s l\'export < /string >

+ < string name = " hint\_delete\_after\_export " > Toutes les transactions export&#233 ; es seront supprimes apr&#233 ; s l\'export < /string >

app/res/values-fr/strings\_xml,4

- < string name = " title\_default\_accounts " > Choisisez les comptes &#224 ; cr&#233 ; er < /string >

+ < string name = " title\_default\_accounts " > Choisisez les comptes &#224 ; cr&#233 ; s < /string >

- < item > Algerian Dinar < /item >

- < item > Argentine Peso < /item >

- < item > Armenian Dram < /item >

- < item > Aruban Florin < /item >

- < item > Australian Dollar < /item >

- < item > Azerbaijanian Manat < /item >

- < item > Bahamian Dollar < /item >

- < item > Bahraini Dinar < /item >

- < item > Baht < /item >

- < item > Balboa < /item >

- < item > Barbados Dollar < /item >

- < item > Belarussian Ruble < /item >

- < item > Belize Dollar < /item >

- < item > Bermudian Dollar < /item >

- < item > Bolivar Fuerte < /item >

- < item > Boliviano < /item >

- < item > Brazilian Real < /item >

- < item > Brunei Dollar < /item >

- < item > Bulgarian Lev < /item >

- < item > Burundi Franc < /item >

- < item > Canadian Dollar < /item >

- < item > Cape Verde Escudo < /item >

- < item > Cayman Islands Dollar < /item >

- < item > CFA Franc BCEAO < /item >

- < item > CFA Franc BEAC < /item >

- < item > CFP Franc < /item >

- < item > Chilean Peso < /item >

- < item > Testing currency < /item >

- < item > Colombian Peso < /item >

- < item > Comoro Franc < /item >

- < item > Congolese Franc < /item >

- < item > Convertible Mark < /item >

- < item > Cordoba Oro < /item >

- < item > Costa Rican Colon < /item >

- < item > Croatian Kuna < /item >

- < item > Cuban Peso < /item >

- < item > Czech Koruna < /item >

- < item > Dalasi < /item >

- < item > Danish Krone < /item >

- < item > Denar < /item >

- < item > Djibouti Franc < /item >

- < item > Dobra < /item >

- < item > Dominican Peso < /item >

- < item > Dong < /item >

- < item > East Caribbean Dollar < /item >

- < item > Egyptian Pound < /item >

- < item > El Salvador Colon < /item >

- < item > Ethiopian Birr < /item >

- < item > Euro < /item >

- < item > Falkland Islands Pound < /item >

- < item > Fiji Dollar < /item >

- < item > Forint < /item >

- < item > Ghana Cedi < /item >

- < item > Gibraltar Pound < /item >

- < item > Gold < /item >

- < item > Gourde < /item >

- < item > Guarani < /item >

- < item > Guinea Franc < /item >

- < item > Guyana Dollar < /item >

- < item > Hong Kong Dollar < /item >

- < item > Hryvnia < /item >

- < item > Iceland Krona < /item >

- < item > Indian Rupee < /item >

- < item > Iranian Rial < /item >

- < item > Iraqi Dinar < /item >

- < item > Jamaican Dollar < /item >

- < item > Jordanian Dinar < /item >

- < item > Kenyan Shilling < /item >

- < item > Kina < /item >

- < item > Kip < /item >

- < item > Kuwaiti Dinar < /item >

- < item > Kwacha < /item >

- < item > Kwanza < /item >

- < item > Kyat < /item >

- < item > Lari < /item >

- < item > Latvian Lats < /item >

- < item > Lebanese Pound < /item >

- < item > Lek < /item >

- < item > Lempira < /item >

- < item > Leone < /item >

- < item > Liberian Dollar < /item >

- < item > Libyan Dinar < /item >

- < item > Lilangeni < /item >

- < item > Lithuanian Litas < /item >

- < item > Loti < /item >

- < item > Malagasy Ariary < /item >

- < item > Malaysian Ringgit < /item >

- < item > Mauritius Rupee < /item >

- < item > Mexican Peso < /item >

- < item > Mexican Unidad de Inversion ( UDI ) < /item >

- < item > Moldovan Leu < /item >

- < item > Moroccan Dirham < /item >

- < item > Mozambique Metical < /item >

- < item > Mvdol < /item >

- < item > Naira < /item >

- < item > Nakfa < /item >

- < item > Namibia Dollar < /item >

- < item > Nepalese Rupee < /item >

- < item > Netherlands Antillean Guilder < /item >

- < item > New Israeli Sheqel < /item >

- < item > New Romanian Leu < /item >

- < item > New Taiwan Dollar < /item >

- < item > New Zealand Dollar < /item >

- < item > Ngultrum < /item >

- < item > North Korean Won < /item >

- < item > Norwegian Krone < /item >

- < item > Nuevo Sol < /item >

- < item > Ouguiya < /item >

- < item > Paanga < /item >

- < item > Pakistan Rupee < /item >

- < item > Palladium < /item >

- < item > Pataca < /item >

- < item > Peso Convertible < /item >

- < item > Peso Uruguayo < /item >

- < item > Philippine Peso < /item >

- < item > Platinum < /item >

- < item > Pound Sterling < /item >

- < item > Pula < /item >

- < item > Qatari Rial < /item >

- < item > Quetzal < /item >

- < item > Rand < /item >

- < item > Rial Omani < /item >

- < item > Riel < /item >

- < item > Rufiyaa < /item >

- < item > Rupiah < /item >

- < item > Russian Ruble < /item >

- < item > Rwanda Franc < /item >

- < item > Saint Helena Pound < /item >

- < item > Saudi Riyal < /item >

- < item > SDR ( Special Drawing Right ) < /item >

- < item > Serbian Dinar < /item >

- < item > Seychelles Rupee < /item >

- < item > Silver < /item >

- < item > Singapore Dollar < /item >

- < item > Solomon Islands Dollar < /item >

- < item > Som < /item >

- < item > Somali Shilling < /item >

- < item > Somoni < /item >

- < item > South Sudanese Pound < /item >

- < item > Sri Lanka Rupee < /item >

- < item > Sucre < /item >

- < item > Sudanese Pound < /item >

- < item > Surinam Dollar < /item >

- < item > Swedish Krona < /item >

- < item > Swiss Franc < /item >

- < item > Syrian Pound < /item >

- < item > Taka < /item >

- < item > Tala < /item >

- < item > Tanzanian Shilling < /item >

- < item > Tenge < /item >

- < item > No currency < /item >

- < item > Trinidad and Tobago Dollar < /item >

- < item > Tugrik < /item >

- < item > Tunisian Dinar < /item >

- < item > Turkish Lira < /item >

- < item > Turkmenistan New Manat < /item >

- < item > UAE Dirham < /item >

- < item > Uganda Shilling < /item >

- < item > UIC- Franc < /item >

- < item > Unidad de Valor Real < /item >

- < item > Unidades de fomento < /item >

- < item > Uruguay Peso en Unidades Indexadas ( URUIURUI ) < /item >

- < item > US Dollar < /item >

- < item > Uzbekistan Sum < /item >

- < item > Vatu < /item >

- < item > WIR Euro < /item >

- < item > WIR Franc < /item >

- < item > Won < /item >

- < item > Yemeni Rial < /item >

- < item > Yen < /item >

- < item > Yuan Renminbi < /item >

- < item > Zambian Kwacha < /item >

- < item > Zimbabwe Dollar < /item >

- < item > Zloty < /item >

+ < item > Algerian Dinar < /item >

+ < item > Argentine Peso < /item >

+ < item > Armenian Dram < /item >

+ < item > Aruban Florin < /item >

+ < item > Australian Dollar < /item >

+ < item > Azerbaijanian Manat < /item >

+ < item > Bahamian Dollar < /item >

+ < item > Bahraini Dinar < /item >

+ < item > Baht < /item >

+ < item > Balboa < /item >

+ < item > Barbados Dollar < /item >

+ < item > Belarussian Ruble < /item >

+ < item > Belize Dollar < /item >

+ < item > Bermudian Dollar < /item >

+ < item > Bolivar Fuerte < /item >

+ < item > Boliviano < /item >

+ < item > Brazilian Real < /item >

+ < item > Brunei Dollar < /item >

+ < item > Bulgarian Lev < /item >

+ < item > Burundi Franc < /item >

+ < item > Canadian Dollar < /item >

+ < item > Cape Verde Escudo < /item >

+ < item > Cayman Islands Dollar < /item >

+ < item > CFA Franc BCEAO < /item >

+ < item > CFA Franc BEAC < /item >

+ < item > CFP Franc < /item >

+ < item > Chilean Peso < /item >

+ < item > Testing currency < /item >

+ < item > Colombian Peso < /item >

+ < item > Comoro Franc < /item >

+ < item > Congolese Franc < /item >

+ < item > Convertible Mark < /item >

+ < item > Cordoba Oro < /item >

+ < item > Costa Rican Colon < /item >

+ < item > Croatian Kuna < /item >

+ < item > Cuban Peso < /item >

+ < item > Czech Koruna < /item >

+ < item > Dalasi < /item >

+ < item > Danish Krone < /item >

+ < item > Denar < /item >

+ < item > Djibouti Franc < /item >

+ < item > Dobra < /item >

+ < item > Dominican Peso < /item >

+ < item > Dong < /item >

+ < item > East Caribbean Dollar < /item >

+ < item > Egyptian Pound < /item >

+ < item > El Salvador Colon < /item >

+ < item > Ethiopian Birr < /item >

+ < item > Euro < /item >

+ < item > Falkland Islands Pound < /item >

+ < item > Fiji Dollar < /item >

+ < item > Forint < /item >

+ < item > Ghana Cedi < /item >

+ < item > Gibraltar Pound < /item >

+ < item > Gold < /item >

+ < item > Gourde < /item >

+ < item > Guarani < /item >

+ < item > Guinea Franc < /item >

+ < item > Guyana Dollar < /item >

+ < item > Hong Kong Dollar < /item >

+ < item > Hryvnia < /item >

+ < item > Iceland Krona < /item >

+ < item > Indian Rupee < /item >

+ < item > Iranian Rial < /item >

+ < item > Iraqi Dinar < /item >

+ < item > Jamaican Dollar < /item >

+ < item > Jordanian Dinar < /item >

+ < item > Kenyan Shilling < /item >

+ < item > Kina < /item >

+ < item > Kip < /item >

+ < item > Kuwaiti Dinar < /item >

+ < item > Kwacha < /item >

+ < item > Kwanza < /item >

+ < item > Kyat < /item >

+ < item > Lari < /item >

+ < item > Latvian Lats < /item >

+ < item > Lebanese Pound < /item >

+ < item > Lek < /item >

+ < item > Lempira < /item >

+ < item > Leone < /item >

+ < item > Liberian Dollar < /item >

+ < item > Libyan Dinar < /item >

+ < item > Lilangeni < /item >

+ < item > Lithuanian Litas < /item >

+ < item > Loti < /item >

+ < item > Malagasy Ariary < /item >

+ < item > Malaysian Ringgit < /item >

+ < item > Mauritius Rupee < /item >

+ < item > Mexican Peso < /item >

+ < item > Mexican Unidad de Inversion ( UDI ) < /item >

+ < item > Moldovan Leu < /item >

+ < item > Moroccan Dirham < /item >

+ < item > Mozambique Metical < /item >

+ < item > Mvdol < /item >

+ < item > Naira < /item >

+ < item > Nakfa < /item >

+ < item > Namibia Dollar < /item >

+ < item > Nepalese Rupee < /item >

+ < item > Netherlands Antillean Guilder < /item >

+ < item > New Israeli Sheqel < /item >

+ < item > New Romanian Leu < /item >

+ < item > New Taiwan Dollar < /item >

+ < item > New Zealand Dollar < /item >

+ < item > Ngultrum < /item >

+ < item > North Korean Won < /item >

+ < item > Norwegian Krone < /item >

+ < item > Nuevo Sol < /item >

+ < item > Ouguiya < /item >

+ < item > Paanga < /item >

+ < item > Pakistan Rupee < /item >

+ < item > Palladium < /item >

+ < item > Pataca < /item >

+ < item > Peso Convertible < /item >

+ < item > Peso Uruguayo < /item >

+ < item > Philippine Peso < /item >

+ < item > Platinum < /item >

+ < item > Pound Sterling < /item >

+ < item > Pula < /item >

+ < item > Qatari Rial < /item >

+ < item > Quetzal < /item >

+ < item > Rand < /item >

+ < item > Rial Omani < /item >

+ < item > Riel < /item >

+ < item > Rufiyaa < /item >

+ < item > Rupiah < /item >

+ < item > Russian Ruble < /item >

+ < item > Rwanda Franc < /item >

+ < item > Saint Helena Pound < /item >

+ < item > Saudi Riyal < /item >

+ < item > SDR ( Special Drawing Right ) < /item >

+ < item > Serbian Dinar < /item >

+ < item > Seychelles Rupee < /item >

+ < item > Silver < /item >

+ < item > Singapore Dollar < /item >

+ < item > Solomon Islands Dollar < /item >

+ < item > Som < /item >

+ < item > Somali Shilling < /item >

+ < item > Somoni < /item >

+ < item > South Sudanese Pound < /item >

+ < item > Sri Lanka Rupee < /item >

+ < item > Sucre < /item >

+ < item > Sudanese Pound < /item >

+ < item > Surinam Dollar < /item >

+ < item > Swedish Krona < /item >

+ < item > Swiss Franc < /item >

+ < item > Syrian Pound < /item >

+ < item > Taka < /item >

+ < item > Tala < /item >

+ < item > Tanzanian Shilling < /item >

+ < item > Tenge < /item >

+ < item > No currency < /item >

+ < item > Trinidad and Tobago Dollar < /item >

+ < item > Tugrik < /item >

+ < item > Tunisian Dinar < /item >

+ < item > Turkish Lira < /item >

+ < item > Turkmenistan New Manat < /item >

+ < item > UAE Dirham < /item >

+ < item > Uganda Shilling < /item >

+ < item > UIC- Franc < /item >

+ < item > Unidad de Valor Real < /item >

+ < item > Unidades de fomento < /item >

+ < item > Uruguay Peso en Unidades Indexadas ( URUIURUI ) < /item >

+ < item > US Dollar < /item >

+ < item > Uzbekistan Sum < /item >

+ < item > Vatu < /item >

+ < item > WIR Euro < /item >

+ < item > WIR Franc < /item >

+ < item > Won < /item >

+ < item > Yemeni Rial < /item >

+ < item > Yen < /item >

+ < item > Yuan Renminbi < /item >

+ < item > Zambian Kwacha < /item >

+ < item > Zimbabwe Dollar < /item >

+ < item > Zloty < /item >

- < string- array name = " default\_accounts " >

- < item > D&#233 ; penses < /item >

- < item > Revenus < /item >

- < item > Actifs < /item >

- < item > Capitaux propres < /item >

- < item > Charges < /item >

- < /string- array >

- < string name = " error\_no\_accounts " > Aucun compte &#233 ; xistant dans Gnucash.\nCr&#233 ; er un compte avant d\'ajouter un widget < /string >

- < string name = " title\_build\_version " > Version logiciel < /string >

- < string name = " title\_license " > Licence < /string >

- < string name = " summary\_licence\_details " > Apache License v2.0. Clickez pour plus de d&#233 ; tails < /string >

- < string name = " title\_general\_prefs " > G&#233 ; n&#233 ; ral < /string >

- < string name = " label\_widget\_configuration " > S&#233 ; lectionnez un compte < /string >

- < string name = " toast\_no\_transactions\_to\_export " > Il n\'existe pas de transactions disponible pour l\'exportation < /string >

- < string name = " title\_about\_gnucash " > A propos de Gnucash < /string >

- < string name = " summary\_about\_gnucash " > GnucashMobile est une application mobile de suivis de finance pour Android.\nElle permet un suivi simple des d&#233 ; penses qui peut &#234 ; tre export&#233 ; vers le format Open Financial eXchange ( OFX ) et import&#233 ; dans Gnucash pour PC. < /string >

- < string name = " title\_about " > A propos < /string >

- < string name = " toast\_error\_exporting " > Impossible d\'&#233 ; crire un fichier OFX en:\n < /string >

- < string name = " toast\_ofx\_exported\_to " > Fichier OFX export&#233 ; en:\n < /string >

- < string name = " title\_export\_email " > Gnucash OFX export < /string >

- < string name = " description\_export\_email " > Gnucash OFX Export de < /string >

- < string name = " header\_transaction\_settings " > Transactions < /string >

- < string name = " title\_transaction\_preferences " > Pr&#233 ; f&#233 ; rences des transactions < /string >

- < string name = " title\_account\_preferences " > Pr&#233 ; f&#233 ; rences du compte < /string >

- < string name = " title\_default\_transaction\_type " > Type de transaction par d&#233 ; faut < /string >

- < string name = " summary\_default\_transaction\_type " > Le type de transaction &#224 ; utiliser par d&#233 ; faut , CREDIT ou DEBIT < /string >

- < string- array name = " transaction\_types " >

- < item > CREDIT < /item >

- < item > DEBIT < /item >

- < /string- array >

- < string name = " delete\_all\_transactions\_confirmation\_message " > Etes vous s&#251 ; re de vouloir supprimer TOUTES les transactions? < /string >

- < string name = " delete\_transaction\_confirmation\_message " > Etes vous s&#251 ; re de vouloir supprimer cette transaction? < /string >

- < string name = " title\_export\_preference\_category " > Export < /string >

- < string name = " title\_export\_all\_transactions " > Exporter toutes les transactions < /string >

- < string name = " title\_always\_delete\_exported\_transactions " > Supprimer les transactions export&#233 ; s < /string >

- < string name = " title\_default\_export\_email " > Email d\'export par d&#233 ; faut < /string >

- < string name = " summary\_default\_export\_email " > Email par d&#233 ; faut pour les exports. Vous pourrez toujours le changer lors de votre prochain export. < /string >

- < string name = " label\_double\_entry\_account " > Transfert entre compte < /string >

- < string name = " summary\_use\_double\_entry " > Toutes les transactions seront transf&#233 ; rer d\'un compte &#224 ; l\'autre < /string >

- < string name = " title\_use\_double\_entry " > Activer les doubles entr&#233 ; es < /string >

- < string name = " account\_balance " > Balance < /string >

- < string name = " toast\_no\_account\_name\_entered " > Enter an account name to create an account < /string >

- < string name = " label\_account\_currency " > Monnaie < /string >

- < string name = " label\_parent\_account " > Compte parent < /string >

- < string name = " title\_xml\_ofx\_header " > Utiliser l\'ent&#234 ; te XML d\'OFX < /string >

- < string name = " summary\_xml\_ofx\_header " > Activez cette option lors d\'un export vers une application tierce autre que GnuCash pour PC < /string >

- < string name = " title\_whats\_new " > Quoi de nouveau < /string >

+ < string- array name = " default\_accounts " >

+ < item > D&#233 ; penses < /item >

+ < item > Revenus < /item >

+ < item > Actifs < /item >

+ < item > Capitaux propres < /item >

+ < item > Charges < /item >

+ < /string- array >

+ < string name = " error\_no\_accounts " > Aucun compte &#233 ; xistant dans Gnucash.\nCr&#233 ; ez un compte avant d\'ajouter un widget < /string >

+ < string name = " title\_build\_version " > Version logiciel < /string >

+ < string name = " title\_license " > Licence < /string >

+ < string name = " summary\_licence\_details " > Apache License v2.0. Clickez pour plus de d&#233 ; tails < /string >

+ < string name = " title\_general\_prefs " > G&#233 ; n&#233 ; ral < /string >

+ < string name = " label\_widget\_configuration " > S&#233 ; lectionner un compte < /string >

+ < string name = " toast\_no\_transactions\_to\_export " > Il n\'existe pas de transactions disponible pour l\'exportation < /string >

+ < string name = " title\_about\_gnucash " > A propos de Gnucash < /string >

+ < string name = " summary\_about\_gnucash " > GnucashMobile est une application mobile de suivis de finance pour Android.\nElle permet un suivi simple des d&#233 ; penses qui peut &#234 ; tre export&#233 ; vers le format Open Financial eXchange ( OFX ) et import&#233 ; dans Gnucash pour PC. < /string >

+ < string name = " title\_about " > A propos < /string >

+ < string name = " toast\_error\_exporting " > Impossible d\'&#233 ; crire un fichier OFX en:\n < /string >

+ < string name = " toast\_ofx\_exported\_to " > Fichier OFX export&#233 ; en:\n < /string >

+ < string name = " title\_export\_email " > Gnucash OFX export < /string >

+ < string name = " description\_export\_email " > Gnucash OFX Export de < /string >

+ < string name = " header\_transaction\_settings " > Transactions < /string >

+ < string name = " title\_transaction\_preferences " > Pr&#233 ; f&#233 ; rences des transactions < /string >

+ < string name = " title\_account\_preferences " > Pr&#233 ; f&#233 ; rences du compte < /string >

+ < string name = " title\_default\_transaction\_type " > Type de transaction par d&#233 ; faut < /string >

+ < string name = " summary\_default\_transaction\_type " > Le type de transaction &#224 ; utiliser par d&#233 ; faut , CREDIT ou DEBIT < /string >

+ < string- array name = " transaction\_types " >

+ < item > CREDIT < /item >

+ < item > DEBIT < /item >

+ < /string- array >

+ < string name = " delete\_all\_transactions\_confirmation\_message " > Etes vous s&#251 ; re de vouloir supprimer TOUTES les transactions? < /string >

+ < string name = " delete\_transaction\_confirmation\_message " > Etes vous s&#251 ; re de vouloir supprimer cette transaction? < /string >

+ < string name = " title\_export\_preference\_category " > Export < /string >

+ < string name = " title\_export\_all\_transactions " > Exporter toutes les transactions < /string >

+ < string name = " title\_always\_delete\_exported\_transactions " > Supprimer les transactions export&#233 ; es < /string >

+ < string name = " title\_default\_export\_email " > Email d\'export par d&#233 ; faut < /string >

+ < string name = " summary\_default\_export\_email " > Email par d&#233 ; faut pour les exports. Vous pourrez toujours le changer lors de votre prochain export. < /string >

+ < string name = " label\_double\_entry\_account " > Transfert entre comptes < /string >

+ < string name = " summary\_use\_double\_entry " > Toutes les transactions seront transf&#233 ; res d\'un compte &#224 ; l\'autre < /string >

+ < string name = " title\_use\_double\_entry " > Activer les doubles entr&#233 ; e < /string >

+ < string name = " account\_balance " > Balance < /string >

+ < string name = " toast\_no\_account\_name\_entered " > Entrer un nom de compte pour crer un compte < /string >

+ < string name = " label\_account\_currency " > Monnaie < /string >

+ < string name = " label\_parent\_account " > Compte parent < /string >

+ < string name = " title\_xml\_ofx\_header " > Utiliser l\'ent&#234 ; te XML d\'OFX < /string >

+ < string name = " summary\_xml\_ofx\_header " > Activez cette option lors d\'un export vers une application tierce autre que GnuCash pour PC < /string >

+ < string name = " title\_whats\_new " > Nouveautes < /string >

- - Mark favorite accounts and quickly access them\n

- - Display different tabs for recent , favorite and all accounts\n

- - Add , view and delete recurring transactions ( daily , weekly , monthly ) \n

- - Mark accounts as placeholder accounts ( cannot contain transactions ) \n

- - Set a default transfer account for each account\n

- - Color code accounts and themed account views\n

- - Create default GnuCash account structure from within app\n

- - Numerous bug fixes and improvements\n

+ - Marquer vos comptes et accdez- y facilement\n

+ - Affichage en onglets pour les comptes rcents , favoris et tous\n

+ - Ajouter , afficher et supprimer des oprations rcurrentes ( quotidien , hebdomadaire , mensuel ) \n

+ - Marque les comptes comme espace rserv ( ne peut pas contenir de transactions ) \n

+ - Dfinir un compte de transfert par dfaut pour chaque compte\n

+ - Code couleur pour comptes et utilisation de thmes\n

+ - Cration d\'une structure de compte GnuCash par dfaut partir de l\'application\n

+ - Nombreuses corrections de bogues et amliorations\n

- < string name = " label\_dismiss " > Passer < /string >

+ < string name = " label\_dismiss " > Passer < /string >

app/src/main/res/drawable-hdpi/ic\_google\_drive\_png,8

- if ( data != null ) {

+ if ( data != null ) {

app/src/main/res/drawable-hdpi/ic\_google\_drive\_png,5

- public void importMostRecentBackup () {

+ public void importMostRecentBackup () {

- if ( backupFile == null ) {

+ if ( backupFile == null ) {

app/src/main/res/drawable-hdpi/ic\_google\_drive\_png,7

- switch ( requestCode ) {

+ switch ( requestCode ) {

app/src/main/res/drawable-hdpi/ic\_google\_drive\_png,6

- if ( resultCode == Activity.RESULT\_CANCELED ) {

+ if ( resultCode == Activity.RESULT\_CANCELED ) {

app/src/org/gnucash/android/model/Money\_java,12

- private void init () {

+ private void init () {

app/src/org/gnucash/android/model/Money\_java,15

- private void setCurrency ( Currency currency ) {

+ private void setCurrency ( @NonNull Currency currency ) {

integration-tests/src/org/gnucash/android/test/unit/MoneyTest\_java,1

- private Money money ;

+ private Money money ;

app/src/main/java/org/gnucash/android/receivers/TransactionRecorder\_java,2

- Split split = new Split ( amount.absolute () , accountUID ) ;

+ Split split = new Split ( amount.abs () , accountUID ) ;

GnucashMobile/res/layout/fragment\_transactions\_list\_xml,1

- android:layout\_width = " fill\_parent "

- android:layout\_height = " 0dip "

+ android:layout\_width = " match\_parent "

+ android:layout\_height = " 0dp "

app/src/main/res/values-pt/strings\_xml,1

- < string name = " label\_create\_account " > Criar Conta < /string >

+ < string name = " title\_create\_account " > Criar Conta < /string >

app/src/main/res/drawable-hdpi/ic\_arrow\_drop\_down\_white\_24dp\_png,2

- public int getCalculatorKeysLayout () {

+ public @XmlRes int getCalculatorKeysLayout () {

- public void setCalculatorKeysLayout ( @XmlRes int mCalculatorKeysLayout ) {

- this.mCalculatorKeysLayout = mCalculatorKeysLayout ;

+ public void setCalculatorKeysLayout ( @XmlRes int calculatorKeysLayout ) {

+ this.mCalculatorKeysLayout = calculatorKeysLayout ;

app/src/main/res/drawable-hdpi/ic\_check\_white\_24dp\_png,7

- int index = mPager.getCurrentItem () ;

+ int index = mViewPager.getCurrentItem () ;

app/src/main/res/layout/dialog\_account\_delete\_xml,1

- < include android:id = " @+ id/transactions\_options " layout = " @layout/radio\_group\_delete\_or\_move " / >

+ < include android:id = " @+ id/transactions\_options " layout = " @layout/radio\_group\_delete\_or\_move " / >

app/src/org/gnucash/android/ui/widget/WidgetConfigurationActivity\_java,1

- Cursor cursor = mAccountsDbAdapter.fetchAllAccounts () ;

+ Cursor cursor = mAccountsDbAdapter.fetchAllRecords () ;

app/res/values/strings\_xml,1

- < string name = " app\_version\_name " > 1.3.2 < /string >

+ < string name = " app\_version\_name " > 1.3.3 < /string >

app/src/main/java/org/gnucash/android/ui/transaction/TransactionDetailActivity\_java,1

- accountName.setText ( accountsDbAdapter.getAccountName ( split.getAccountUID () ) ) ;

+ accountName.setText ( accountsDbAdapter.getAccountFullName ( split.getAccountUID () ) ) ;

app/src/main/java/org/gnucash/android/ui/account/AccountFormFragment\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

- import org.gnucash.android.db.CommoditiesDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.CommoditiesDbAdapter ;

app/src/org/gnucash/android/ui/colorpicker/ColorPickerDialog\_java,3

- private void initializeColorSquarePreview ( String colorHex ) {

+ private void initializeColorSquarePreview ( @NonNull String colorHex ) {

app/src/org/gnucash/android/ui/accounts/AccountsListFragment\_java,3

- Cursor cursor = ( ( AccountsDbAdapter ) mDatabaseAdapter ) .fetchAllAccounts () ;

+ Cursor cursor = ( ( AccountsDbAdapter ) mDatabaseAdapter ) .fetchAllRecords () ;

app/src/main/java/org/gnucash/android/db/adapter/RecurrenceDbAdapter\_java,11

- return fetchAllRecords ( null , null ) ;

+ return fetchAllRecords ( null , null , null ) ;

- public Cursor fetchAllRecords ( String where , String[] whereArgs ) {

- return mDb.query ( mTableName , null , where , whereArgs , null , null , null ) ;

+ public Cursor fetchAllRecords ( String where , String[] whereArgs , String orderBy ) {

+ return mDb.query ( mTableName , null , where , whereArgs , null , null , orderBy ) ;

app/src/main/java/org/gnucash/android/db/adapter/RecurrenceDbAdapter\_java,10

- protected ContentValues populateBaseModelAttributes ( @NonNull ContentValues contentValues , @NonNull Model model ) {

+ protected ContentValues extractBaseModelAttributes ( @NonNull ContentValues contentValues , @NonNull Model model ) {

app/src/main/java/org/gnucash/android/db/adapter/RecurrenceDbAdapter\_java,9

- protected abstract Model buildModelInstance ( @NonNull final Cursor cursor ) ;

+ public abstract Model buildModelInstance ( @NonNull final Cursor cursor ) ;

app/src/main/java/org/gnucash/android/db/adapter/RecurrenceDbAdapter\_java,4

- Cursor cursor = fetchAllRecords ( CommodityEntry.COLUMN\_MNEMONIC + " = ? " , new String[] { currencyCode } ) ;

+ Cursor cursor = fetchAllRecords ( CommodityEntry.COLUMN\_MNEMONIC + " = ? " , new String[] { currencyCode } , null ) ;

app/src/main/res/layout/account\_spinner\_item\_xml,1

- android:textSize = " 20sp "

+ android:textSize = " 18sp "

app/src/main/res/drawable-hdpi/ic\_star\_border\_white\_48dp\_png,4

- mPager.setCurrentItem ( index ) ;

+ mViewPager.setCurrentItem ( index ) ;

integration-tests/src/org/gnucash/android/test/unit/AccountTest\_java,2

- assertTrue ( uid.contains ( " tele- pathy " ) ) ;

+ assertTrue ( uid.contains ( " telepathy " ) ) ;

GnucashMobile/src/org/gnucash/android/db/DatabaseHelper\_java,1

- private static final int DATABASE\_VERSION = 1 ;

+ private static final int DATABASE\_VERSION = 2 ;

app/res/menu/account\_actions\_xml,1

- android:orderInCategory = " 3 " / >

+ android:orderInCategory = " 3 " / >

README\_md,1

- < string name = " summary\_about\_gnucash " > GnucashMobile es una aplicaci&#243 ; n de registro de gastos financieros para Android.\nPermite un registro flexible de gastos en cualquier lugar que luego pueden ser exportados al formato Open Financial eXchange ( OFX ) e importados en la aplicaci&#243 ; n Gnucash para el escritorio. < /string >

+ < string name = " summary\_about\_gnucash " > GnucashMobile es una aplicaci&#243 ; n de registro de gastos financieros para Android.\nPermite un registro flexible de gastos en cualquier lugar que luego pueden ser exportados al formato Open Financial eXchange ( OFX ) e importados en la aplicaci&#243 ; n Gnucash para el escritorio. < /string >

app/src/main/res/layout/fragment\_scheduled\_events\_list\_xml,1

- android:listSelector = " @drawable/abs\_\_list\_selector\_holo\_light "

+ android:listSelector = " @drawable/abc\_list\_selector\_holo\_light "

app/src/main/java/org/gnucash/android/model/Transaction\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

app/src/main/java/org/gnucash/android/model/Transaction\_java,3

- absAmount = split.getValue () .absolute () ;

+ absAmount = split.getValue () .abs () ;

- absAmount = split.getQuantity () .absolute () ;

+ absAmount = split.getQuantity () .abs () ;

app/src/main/java/org/gnucash/android/model/Transaction\_java,2

- Money amount = split.getValue () .absolute () ;

+ Money amount = split.getValue () .abs () ;

app/src/main/res/layout/fragment\_bar\_chart\_xml,1

- android:minHeight = " 400dp "

+ android:minHeight = " 300dp "

app/src/main/java/org/gnucash/android/ui/export/ExportFormFragment\_java,4

- RecurrencePickerDialog.OnRecurrenceSetListener ,

- CalendarDatePickerDialog.OnDateSetListener ,

- RadialTimePickerDialog.OnTimeSetListener {

+ RecurrencePickerDialogFragment.OnRecurrenceSetListener ,

+ CalendarDatePickerDialogFragment.OnDateSetListener ,

+ RadialTimePickerDialogFragment.OnTimeSetListener {

app/src/org/gnucash/android/ui/settings/DeleteAllTransacationsConfirmationDialog\_java,2

- public class DeleteAccountsConfirmationDialog extends DialogFragment {

+ public class DeleteAllAccountsConfirmationDialog extends DialogFragment {

- public static DeleteAccountsConfirmationDialog newInstance () {

- DeleteAccountsConfirmationDialog frag = new DeleteAccountsConfirmationDialog () ;

+ public static DeleteAllAccountsConfirmationDialog newInstance () {

+ DeleteAllAccountsConfirmationDialog frag = new DeleteAllAccountsConfirmationDialog () ;

app/res/drawable-hdpi/navigation\_next\_item\_png,7

- AccountType type = ( AccountType ) ( ( Spinner ) findViewById ( R.id.chart\_data\_spinner ) ) .getSelectedItem () ;

- mRenderer.setChartTitle ( type.toString () ) ;

-

- setDataset ( type , false ) ;

+ mAccountType = ( AccountType ) ( ( Spinner ) findViewById ( R.id.chart\_data\_spinner ) ) .getSelectedItem () ;

+ mRenderer.setChartTitle ( mAccountType.toString () ) ;

+ setDataset ( false ) ;

GnucashMobile/res/layout/list\_item\_account\_xml,2

- android:text = " @string/label\_account\_name "

+ android:text = " @string/label\_account\_name "

app/src/test/java/org/gnucash/android/test/unit/export/BackupTest\_java,1

- import org.gnucash.android.test.unit.util.GnucashTestRunner ;

- import org.gnucash.android.test.unit.util.ShadowCrashlytics ;

- import org.gnucash.android.test.unit.util.ShadowUserVoice ;

+ import org.gnucash.android.test.unit.testutil.GnucashTestRunner ;

+ import org.gnucash.android.test.unit.testutil.ShadowCrashlytics ;

+ import org.gnucash.android.test.unit.testutil.ShadowUserVoice ;

app/src/main/res/layout-sw600dp/activity\_settings\_xml,3

- String typePref = PreferenceManager.getDefaultSharedPreferences ( getActivity () ) .getString ( getString ( R.string.key\_default\_transaction\_type ) , " DEBIT " ) ;

+ String typePref = PreferenceActivity.getBookSharedPreferences ( getActivity () ) .getString ( getString ( R.string.key\_default\_transaction\_type ) , " DEBIT " ) ;

app/src/main/res/layout-sw600dp/activity\_settings\_xml,7

- public Dialog onCreateDialog ( Bundle savedInstanceState ) {

+ @NonNull public Dialog onCreateDialog ( Bundle savedInstanceState ) {

app/res/xml/fragment\_transaction\_preferences\_xml,1

- < ListPreference android:title = " @string/title\_default\_transaction\_type " android:entryValues = " @array/key\_transaction\_type\_values " android:entries = " @array/transaction\_types " android:key = " @string/key\_default\_transaction\_type " android:summary = " @string/summary\_default\_transaction\_type " / >

- < CheckBoxPreference android:summary = " @string/summary\_use\_double\_entry " android:key = " @string/key\_use\_double\_entry " android:title = " @string/title\_use\_double\_entry " / >

+ < ListPreference android:title = " @string/title\_default\_transaction\_type "

+ android:entryValues = " @array/key\_transaction\_type\_values "

+ android:entries = " @array/transaction\_types "

+ android:key = " @string/key\_default\_transaction\_type "

+ android:summary = " @string/summary\_default\_transaction\_type " / >

+ < CheckBoxPreference android:summary = " @string/summary\_use\_double\_entry "

+ android:key = " @string/key\_use\_double\_entry "

+ android:title = " @string/title\_use\_double\_entry " / >

GnucashMobile/res/values-es/strings\_xml,1

- < string name = " summary\_default\_transaction\_type " > El tipo de transacci&#243 ; n que usar por defecto: ABONO o CARGO < /string >

+ < string name = " summary\_default\_transaction\_type " > El tipo de transacci&#243 ; n a utilizar por defecto: ABONO o CARGO < /string >

app/src/org/gnucash/android/importer/ImportAsyncTask\_java,2

- public ImportAsyncTask ( Activity context , TaskDelegate delegate ) {

+ public ImportAsyncTask ( Activity context , @Nullable TaskDelegate delegate ) {

app/src/main/java/org/gnucash/android/receivers/AccountCreator\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

app/src/main/java/org/gnucash/android/ui/transaction/dialog/TimePickerDialogFragment\_java,1

- import java.util.Calendar ;

- import java.util.GregorianCalendar ;

-

+ import java.util.Calendar ;

+ import java.util.GregorianCalendar ;

+

app/src/main/res/layout/activity\_pie\_chart\_xml,1

- < com.github.mikephil.charting.charts.PieChart android:id = " @+ id/chart "

+ < com.github.mikephil.charting.charts.PieChart android:id = " @+ id/pie\_chart "

app/src/main/java/org/gnucash/android/ui/colorpicker/ColorPickerDialog\_java,1

- public class ColorPickerDialog extends SherlockDialogFragment implements OnColorSelectedListener {

+ public class ColorPickerDialog extends DialogFragment implements OnColorSelectedListener {

GnucashMobileTest/src/org/gnucash/android/test/unit/MoneyTest\_java,1

- assertEquals ( " 15 , 75 " , money.formattedString ( Locale.GERMAN ) ) ;

- assertEquals ( " 15.75 " , money.formattedString ( Locale.US ) ) ;

+

+ assertEquals ( " 15 , 75 \u20AC " , money.formattedString ( Locale.GERMAN ) ) ;

+ assertEquals ( " 15.75 \u20AC " , money.formattedString ( Locale.US ) ) ;

app/res/xml-v11/fragment\_transaction\_preferences\_xml,3

- < item > MUTUAL FUND < /item >

+ < item > MUTUAL\_FUND < /item >

app/res/xml-v11/fragment\_transaction\_preferences\_xml,2

- < item > MUTUAL\_FUND < /item >

+ < item > MUTUAL FUND < /item >

- < item > CREDIT CARD < /item >

+ < item > CREDIT\_CARD < /item >

GnucashMobile/src/org/gnucash/android/ui/widget/WidgetConfigurationActivity\_java,2

- Editor editor = PreferenceManager

- .getDefaultSharedPreferences ( context ) .edit () ;

- editor.remove ( TransactionsListFragment.SELECTED\_ACCOUNT\_ID + appWidgetId ) ;

+ Editor editor = PreferenceManager.getDefaultSharedPreferences ( context ) .edit () ;

+ editor.remove ( TransactionsListFragment.SELECTED\_ACCOUNT\_ID + appWidgetId ) ;

+ editor.commit () ;

GnucashMobile/src/org/gnucash/android/ui/TransactionsListFragment\_java,1

+

- private long mAccountUid ;

+ private long accountID ;

- this.mAccountUid = accountID ;

+ this.accountID = accountID ;

- return ( ( TransactionsDbAdapter ) mDatabaseAdapter ) .fetchAllTransactionsForAccount ( mAccountUid ) ;

+ return ( ( TransactionsDbAdapter ) mDatabaseAdapter ) .fetchAllTransactionsForAccount ( accountID ) ;

app/src/main/java/org/gnucash/android/export/ExportFormat\_java,1

- GNC\_XML ( " GnuCash XML " ) ;

+ XML ( " GnuCash XML " ) ;

app/src/main/java/org/gnucash/android/export/ExportFormat\_java,2

- private void bindViews () {

+ private void bindViews () {

app/src/main/java/org/gnucash/android/export/ExportFormat\_java,4

- if ( defaultExportFormat.equalsIgnoreCase ( ExportFormat.GNC\_XML.name () ) ) {

+ if ( defaultExportFormat.equalsIgnoreCase ( ExportFormat.XML.name () ) ) {

app/src/org/gnucash/android/export/ExporterTask\_java,5

- + " /gnucash/ " + ExportDialogFragment.buildExportFilename ( mExportParams.getExportFormat () ) ) ;

+ + " /gnucash/ " + Exporter.buildExportFilename ( mExportParams.getExportFormat () ) ) ;

app/src/main/res/drawable-hdpi/ic\_action\_backspace\_png,5

- } else {

- accountsDbAdapter.recursiveDeleteAccount ( accountsDbAdapter.getID ( mOriginAccountUID ) ) ;

+

+

+ accountsDbAdapter.recursiveDeleteAccount ( accountsDbAdapter.getID ( mOriginAccountUID ) ) ;

+

app/src/main/java/org/gnucash/android/service/ScheduledActionService\_java,2

- Log.e ( LOG\_TAG , " Scheduled action with UID " + actionUID + " could not be found in the db with path " + db.getPath () ) ;

+ Log.e ( LOG\_TAG , " Scheduled transaction with UID " + actionUID + " could not be found in the db with path " + db.getPath () ) ;

GnucashMobileTest/src/org/gnucash/android/test/TransactionsFragmentTest\_java,3

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_TRANSACTIONS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_TRANSACTIONS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/TransactionsFragmentTest\_java,4

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

app/src/org/gnucash/android/db/MigrationHelper\_java,1

- GncXmlExporter exporter = new GncXmlExporter ( exportParams , db ) ;

+ GncXmlExporter exporter = new GncXmlExporter ( exportParams ) ;

GnucashMobile/src/org/gnucash/android/ui/AccountsActivity\_java,2

- public class AccountsActivity extends SherlockFragmentActivity implements OnAccountSelectedListener {

+ public class AccountsActivity extends SherlockFragmentActivity implements OnItemClickedListener {

app/src/main/java/org/gnucash/android/ui/passcode/KeyboardFragment\_java,2

- public void onPasscodeEntered ( String pass ) ;

+ void onPasscodeEntered ( String pass ) ;

app/src/main/java/org/gnucash/android/model/Account\_java,1

- private List < Transaction > mTransactionsList = new ArrayList < Transaction > () ;

+ private List < Transaction > mTransactionsList = new ArrayList < > () ;

GnucashMobile/src/org/gnucash/android/ui/AccountsListFragment\_java,1

- private ActionMode.Callback mActionModCallbacks = new Callback () {

+ private ActionMode.Callback mActionModeCallbacks = new Callback () {

- mode.setTitle ( " 1 account selected " ) ;

+ mode.setTitle ( " 1 selected " ) ;

GnucashMobile/src/org/gnucash/android/ui/AccountsListFragment\_java,5

- public void onDestroyView () {

+ public void onDestroy () {

GnucashMobile/src/org/gnucash/android/ui/AccountsListFragment\_java,4

- mActionMode = getSherlockActivity () .startActionMode ( mActionModCallbacks ) ;

+ mActionMode = getSherlockActivity () .startActionMode ( mActionModeCallbacks ) ;

app/res/raw-de-rCH/default\_accounts\_gnucash,1

- < version > 1.2.7- SNAPSHOT < /version >

+ < version > 1.3.0- SNAPSHOT < /version >

app/src/test/java/org/gnucash/android/test/unit/util/PreferencesHelperTest\_java,3

- package org.gnucash.android.test.unit.util ;

+ package org.gnucash.android.test.unit.testutil ;

app/src/test/java/org/gnucash/android/test/unit/util/PreferencesHelperTest\_java,2

- package org.gnucash.android.test.unit.util ;

+ package org.gnucash.android.test.unit.testutil ;

app/src/test/java/org/gnucash/android/test/unit/util/PreferencesHelperTest\_java,4

- package org.gnucash.android.test.unit.util ;

+ package org.gnucash.android.test.unit.testutil ;

app/src/main/java/org/gnucash/android/ui/util/CursorRecyclerAdapter\_java,1

- \* @return Returns the previously set Cursor , or null if there wasa not one.

+ \* @return Returns the previously set Cursor , or null if there was not one.

GnucashMobile/res/layout/fragment\_account\_detail\_xml,1

- android:text = " @string/add\_account " / >

+ android:text = " @string/title\_add\_account " / >

app/src/test/java/org/gnucash/android/test/unit/model/TransactionTest\_java,1

- import org.gnucash.android.test.unit.util.GnucashTestRunner ;

- import org.gnucash.android.test.unit.util.ShadowCrashlytics ;

- import org.gnucash.android.test.unit.util.ShadowUserVoice ;

+ import org.gnucash.android.test.unit.testutil.GnucashTestRunner ;

+ import org.gnucash.android.test.unit.testutil.ShadowCrashlytics ;

+ import org.gnucash.android.test.unit.testutil.ShadowUserVoice ;

GnucashMobile/res/values/dimens\_xml,1

- < dimen name = " widget\_margin " > 8dp < /dimen >

+ < dimen name = " widget\_margin " > 8dp < /dimen >

GnucashMobile/src/org/gnucash/android/ui/transactions/TransactionsListFragment\_java,1

- private static final String SAVED\_SELECTED\_ITEMS = " selected\_items " ;

- public static final String SELECTED\_ACCOUNT\_ID = " selected\_account\_id " ;

+ private static final String SAVED\_SELECTED\_ITEMS = " selected\_items " ;

+ public static final String SELECTED\_ACCOUNT\_ID = " selected\_account\_id " ;

app/src/org/gnucash/android/ui/settings/AboutPreferenceFragment\_java,4

- public static class AccountImporterTask extends AsyncTask < Uri , Void , Boolean > {

+ public static class AccountImporterTask extends AsyncTask < InputStream , Void , Boolean > {

app/src/main/res/layout/activity\_accounts\_xml,1

- android:layout\_margin = " 12dp "

+ android:layout\_margin = " 20dp "

app/src/org/gnucash/android/ui/transaction/dialog/BulkMoveDialogFragment\_java,3

- trxnAdapter.moveTranscation ( trxnAdapter.getUID ( trxnId ) , srcAccountUID , dstAccountUID ) ;

+ trxnAdapter.moveTransaction ( trxnAdapter.getUID ( trxnId ) , srcAccountUID , dstAccountUID ) ;

app/src/main/res/layout/dialog\_bulk\_move\_xml,2

- android:layout\_margin = " @dimen/dialog\_padding " / >

+ android:layout\_marginBottom = " @dimen/dialog\_padding " / >

app/src/androidTest/java/org/gnucash/android/test/ui/FirstRunWizardActivityTest\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

- import org.gnucash.android.db.SplitsDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.SplitsDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

integration-tests/src/main/AndroidManifest\_xml,1

- < uses- sdk android:minSdkVersion = " 8 " android:targetSdkVersion = " 16 " / >

+ < uses- sdk android:minSdkVersion = " 8 " android:targetSdkVersion = " 18 " / >

app/src/main/java/org/gnucash/android/ui/transaction/SplitEditorFragment\_java,1

+ import org.gnucash.android.ui.util.OnTransferFundsListener ;

- import org.gnucash.android.ui.util.OnTransferFundsListener ;

integration-tests/AndroidManifest\_xml,1

- android:versionName = " 1.1.2 " >

+ android:versionName = " 1.2.0 " >

GnucashMobile/src/org/gnucash/android/data/Money\_java,2

- private AccountType mAccountType = AccountType.CHECKING ;

+ private AccountType mAccountType = AccountType.CASH ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,12

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_TRANSACTIONS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_TRANSACTIONS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,3

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,2

- public class AccountsActivityTest extends ActivityInstrumentationTestCase2 < AccountsActivity > {

+ public class AccountsActivityTest extends ActivityInstrumentationTestCase2 < MainActivity > {

- super ( AccountsActivity.class ) ;

+ super ( MainActivity.class ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,5

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,4

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,7

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,6

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_TRANSACTIONS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_TRANSACTIONS\_LIST ) ;

GnucashMobileTest/src/org/gnucash/android/test/AccountsActivityTest\_java,9

- .findFragmentByTag ( AccountsActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

+ .findFragmentByTag ( MainActivity.FRAGMENT\_ACCOUNTS\_LIST ) ;

GnucashMobile/src/org/gnucash/android/ui/transactions/NewTransactionFragment\_java,3

- mAccountsSpinner.setAdapter ( mCursorAdapter ) ;

+ mAccountsSpinner.setAdapter ( mCursorAdapter ) ;

app/src/androidTest/java/org/gnucash/android/test/ui/ExportTransactionsTest\_java,4

- onView ( withText ( " Done " ) ) .perform ( click () ) ;

+ onView ( withText ( " OK " ) ) .perform ( click () ) ;

GnucashMobile/res/values/strings\_xml,1

- < string name = " app\_name " > Gnucash < /string >

+ < string name = " app\_name " > GnuCash < /string >

app/src/main/res/drawable/selected\_background\_xml,1

- < item android:drawable = " @drawable/abs\_\_item\_background\_holo\_light "

+ < item android:drawable = " @drawable/abc\_item\_background\_holo\_light "

- < item android:drawable = " @drawable/abs\_\_item\_background\_holo\_light "

+ < item android:drawable = " @drawable/abc\_item\_background\_holo\_light "

app/src/org/gnucash/android/ui/passcode/PassLockActivity\_java,1

- getIntent () .getLongExtra ( UxArgument.SELECTED\_ACCOUNT\_UID , 0L ) )

+ getIntent () .getStringExtra ( UxArgument.SELECTED\_ACCOUNT\_UID ) )

app/src/org/gnucash/android/ui/transactions/BulkMoveDialogFragment\_java,1

- Cursor cursor = mAccountsDbAdapter.fetchAllAccounts () ;

+ Cursor cursor = mAccountsDbAdapter.fetchAllRecords () ;

app/res/xml/fragment\_about\_preferences\_xml,1

- < Preference android:key = " @string/key\_about " android:summary = " @string/summary\_about\_gnucash " android:title = " @string/title\_about " / > < Preference android:key = " @string/key\_build\_version " android:title = " @string/title\_build\_version " / >

+ < Preference android:key = " @string/key\_about " android:summary = " @string/summary\_about\_gnucash "

+ android:title = " @string/title\_about " / >

+ < Preference android:key = " @string/key\_build\_version " android:title = " @string/title\_build\_version " / >

- < /Preference >

+ < /Preference >

app/src/main/java/org/gnucash/android/ui/account/DeleteAccountDialogFragment\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

- import org.gnucash.android.db.SplitsDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.SplitsDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

app/src/main/java/org/gnucash/android/model/Commodity\_java,1

- import org.gnucash.android.db.CommoditiesDbAdapter ;

+ import org.gnucash.android.db.adapter.CommoditiesDbAdapter ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,10

- xmlSerializer.text ( " n " ) ;

+ xmlSerializer.text ( " n " ) ;

+

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,15

- private void exportCommodity ( XmlSerializer xmlSerializer , List < Currency > currencies ) throws IOException {

+ private void exportCommodities ( XmlSerializer xmlSerializer , List < Currency > currencies ) throws IOException {

- xmlSerializer.attribute ( null , GncXmlHelper.ATTR\_KEY\_VERSION , " 2.0.0 " ) ;

+ xmlSerializer.attribute ( null , GncXmlHelper.ATTR\_KEY\_VERSION , GncXmlHelper.BOOK\_VERSION ) ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,17

- " split " , " trn " , " ts " , " sx " , " recurrence " } ;

+ " split " , " trn " , " ts " , " sx " , " bgt " , " recurrence " } ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,18

- exportCommodity ( xmlSerializer , currencies ) ;

+ exportCommodities ( xmlSerializer , currencies ) ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,3

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_NAME ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_NAME ) ;

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_NAME ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_NAME ) ;

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_TYPE ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_TYPE ) ;

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_TYPE ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_TYPE ) ;

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCOUNT\_COMMODITY ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_COMMODITY ) ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,5

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACT\_SLOTS ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_SLOTS ) ;

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACT\_SLOTS ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_SLOTS ) ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,4

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCOUNT\_COMMODITY ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_COMMODITY ) ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,7

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCOUNT\_COMMODITY ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_COMMODITY ) ;

app/src/main/java/org/gnucash/android/export/xml/GncXmlExporter\_java,6

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_NAME ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_NAME ) ;

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_NAME ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_NAME ) ;

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_TYPE ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_TYPE ) ;

- xmlSerializer.endTag ( null , GncXmlHelper.TAG\_TYPE ) ;

+ xmlSerializer.endTag ( null , GncXmlHelper.TAG\_ACCT\_TYPE ) ;

- xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCOUNT\_COMMODITY ) ;

+ xmlSerializer.startTag ( null , GncXmlHelper.TAG\_ACCT\_COMMODITY ) ;

app/src/org/gnucash/android/ui/accounts/ExportDialogFragment\_java,4

- private void writeToExternalStorage ( Document doc ) throws IOException {

+ private void writeOfxToExternalStorage ( Document doc ) throws IOException {

-

app/AndroidManifest\_xml,1

- android:versionCode = " 28 "

+ android:versionCode = " 29 "

GnucashMobile/src/org/gnucash/android/ui/NewTransactionFragment\_java,1

- mAmountEditText.setText ( Double.toString ( mTransaction.getAmount () \* 10 ) ) ;

+ mAmountEditText.setText ( Double.toString ( mTransaction.getAmount () ) ) ;

app/src/main/java/org/gnucash/android/ui/util/dialog/DateRangePickerDialogFragment\_java,2

- @Bind ( R.id.calendar\_view ) CalendarPickerView mCalendarPickerView ;

- @Bind ( R.id.btn\_save ) Button mDoneButton ;

- @Bind ( R.id.btn\_cancel ) Button mCancelButton ;

+ @BindView ( R.id.calendar\_view ) CalendarPickerView mCalendarPickerView ;

+ @BindView ( R.id.btn\_save ) Button mDoneButton ;

+ @BindView ( R.id.btn\_cancel ) Button mCancelButton ;

app/src/main/java/org/gnucash/android/ui/common/FormActivity\_java,2

- public enum FormType { ACCOUNT , TRANSACTION , EXPORT , SPLIT\_EDITOR }

+ public enum FormType { ACCOUNT , TRANSACTION , EXPORT , SPLIT\_EDITOR , BUDGET }

app/src/androidTest/java/org/gnucash/android/test/ui/AccountsActivityTest\_java,2

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

- import org.gnucash.android.db.SplitsDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.SplitsDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

integration-tests/src/main/java/org/gnucash/android/test/ui/ExportTransactionsTest\_java,1

- mSolo.sleep ( 2000 ) ;

+ mSolo.sleep ( 5000 ) ;

app/src/org/gnucash/android/ui/passcode/PasscodeLockScreenActivity\_java,1

- .putExtra ( UxArgument.SELECTED\_ACCOUNT\_UID , getIntent () .getLongExtra ( UxArgument.SELECTED\_ACCOUNT\_UID , 0L ) )

+ .putExtra ( UxArgument.SELECTED\_ACCOUNT\_UID , getIntent () .getStringExtra ( UxArgument.SELECTED\_ACCOUNT\_UID ) )

app/src/main/java/org/gnucash/android/ui/colorpicker/ColorPickerPalette\_java,1

- int rowMax = ( ( rowNumber + 1 ) \* mNumColumns ) ;

+ int rowMax = ( rowNumber + 1 ) \* mNumColumns ;

app/src/main/java/org/gnucash/android/export/ExportDialogFragment\_java,5

- R.array.export\_destinations , android.R.layout.simple\_spinner\_item ) ;

+ R.array.export\_destinations , android.R.layout.simple\_spinner\_item ) ;

app/src/main/res/layout/cardview\_transaction\_xml,1

- android:layout\_marginRight = " 50dp " / >

+ android:layout\_marginRight = " 52dp " / >

app/src/main/java/org/gnucash/android/ui/wizard/CurrencySelectFragment\_java,1

- import org.gnucash.android.db.CommoditiesDbAdapter ;

+ import org.gnucash.android.db.adapter.CommoditiesDbAdapter ;

app/src/main/res/layout/list\_item\_transaction\_xml,2

- < AutoCompleteTextView

- android:id = " @+ id/input\_transaction\_name "

- android:layout\_width = " match\_parent "

- android:layout\_height = " wrap\_content "

- android:hint = " @string/label\_transaction\_name "

- android:inputType = " textCapSentences "

- android:imeOptions = " actionNext "

- android:nextFocusDown = " @+ id/input\_transaction\_amount "

- android:background = " @android:color/transparent "

- android:textSize = " 22sp " / >

+ < AutoCompleteTextView

+ android:id = " @+ id/input\_transaction\_name "

+ android:layout\_width = " match\_parent "

+ android:layout\_height = " wrap\_content "

+ android:layout\_span = " 2 "

+ android:hint = " @string/label\_transaction\_name "

+ android:inputType = " textCapSentences "

+ android:imeOptions = " actionNext "

+ android:nextFocusDown = " @+ id/input\_transaction\_amount "

+ android:background = " @android:color/transparent "

+ android:textSize = " 22sp " / >

+

app/src/main/res/drawable-hdpi/ic\_close\_black\_24dp\_png,4

- if ( event.getAction () == MotionEvent.ACTION\_UP ) {

- if ( event.getRawX () >= ( mAmountEditText.getRight () - mAmountEditText.getCompoundDrawables () [DRAWABLE\_RIGHT].getBounds () .width () ) ) {

+ if ( event.getAction () == MotionEvent.ACTION\_UP ) {

+ if ( event.getRawX () >= ( mAmountEditText.getRight () - mAmountEditText.getCompoundDrawables () [DRAWABLE\_RIGHT].getBounds () .width () ) ) {

app/src/main/java/org/gnucash/android/export/ExporterAsyncTask\_java,2

- public ExporterAsyncTask ( Activity context ) {

+ public ExporterAsyncTask ( Context context ) {

+ @TargetApi ( 11 )

- mProgressDialog = new ProgressDialog ( mContext ) ;

- mProgressDialog.setTitle ( R.string.title\_progress\_exporting\_transactions ) ;

- mProgressDialog.setIndeterminate ( true ) ;

- mProgressDialog.setProgressStyle ( ProgressDialog.STYLE\_HORIZONTAL ) ;

- if ( Build.VERSION.SDK\_INT > Build.VERSION\_CODES.HONEYCOMB ) {

- mProgressDialog.setProgressNumberFormat ( null ) ;

- mProgressDialog.setProgressPercentFormat ( null ) ;

+ if ( mContext instanceof Activity ) {

+ mProgressDialog = new ProgressDialog ( mContext ) ;

+ mProgressDialog.setTitle ( R.string.title\_progress\_exporting\_transactions ) ;

+ mProgressDialog.setIndeterminate ( true ) ;

+ mProgressDialog.setProgressStyle ( ProgressDialog.STYLE\_HORIZONTAL ) ;

+ if ( Build.VERSION.SDK\_INT > Build.VERSION\_CODES.HONEYCOMB ) {

+ mProgressDialog.setProgressNumberFormat ( null ) ;

+ mProgressDialog.setProgressPercentFormat ( null ) ;

+ }

+ mProgressDialog.show () ;

- mProgressDialog.show () ;

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,49

- public boolean isPlaceholderAccount ( @NonNull String accountUID ) {

+ public boolean isPlaceholderAccount ( String accountUID ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,13

- public void setCurrencyCode ( @NonNull String currencyCode ) {

+ public void setCurrencyCode ( String currencyCode ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,22

- @NonNull

- public static TransactionType getTypeForBalance ( @NonNull AccountType accountType , boolean shouldReduceBalance ) {

+ public static TransactionType getTypeForBalance ( AccountType accountType , boolean shouldReduceBalance ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,23

- public static boolean shouldDecreaseBalance ( @NonNull AccountType accountType , @NonNull TransactionType transactionType ) {

+ public static boolean shouldDecreaseBalance ( AccountType accountType , TransactionType transactionType ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,54

- public int getTransactionMaxSplitNum ( @NonNull String accountUID ) {

+ public int getTransactionMaxSplitNum ( String accountUID ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,16

- public void setDescription ( @NonNull String description ) {

+ public void setDescription ( String description ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,41

- public int getSubAccountCount ( @NonNull String accountUID ) {

+ public int getSubAccountCount ( String accountUID ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,2

- @Override @NonNull

- public Dialog onCreateDialog ( Bundle savedInstanceState ) {

+ @Override public Dialog onCreateDialog ( Bundle savedInstanceState ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,4

- private void initializeColorSquarePreview ( @NonNull String colorHex ) {

+ private void initializeColorSquarePreview ( String colorHex ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,7

- public void setSplits ( List < Split > splitList ) {

+ public void setSplits ( List < Split > splitList ) {

app/src/org/gnucash/android/ui/transaction/dialog/TransactionsDeleteConfirmationDialogFragment\_java,9

- public Money getBalance ( @NonNull String accountUID ) {

+ public Money getBalance ( String accountUID ) {

app/src/main/java/org/gnucash/android/db/DatabaseCursorLoader\_java,2

- protected final ForceLoadContentObserver mObserver ;

+ protected final Loader.ForceLoadContentObserver mObserver ;

app/src/main/res/drawable-hdpi/ic\_dropbox\_android\_png,7

- return GeneralPreferenceFragment.class.getName () .equals ( fragmentName )

+ return BackupPreferenceFragment.class.getName () .equals ( fragmentName )

app/src/main/java/org/gnucash/android/ui/transaction/ScheduledActionsListFragment\_java,1

- inflater.inflate ( R.menu.transactions\_context\_menu , menu ) ;

+ inflater.inflate ( R.menu.schedxactions\_context\_menu , menu ) ;

app/src/org/gnucash/android/data/Transaction\_java,1

- transactionQifBuffer.append ( QifHelper.SPLIT\_AMOUNT\_PREFIX + mAmount.negate () .asString () + newLine ) ;

+ transactionQifBuffer.append ( QifHelper.SPLIT\_AMOUNT\_PREFIX + mAmount.asString () + newLine ) ;

app/src/org/gnucash/android/ui/transactions/TransactionsActivity\_java,1

- Cursor accountsCursor = mAccountsDbAdapter.fetchAllAccounts () ;

+ Cursor accountsCursor = mAccountsDbAdapter.fetchAllRecords () ;

app/src/org/gnucash/android/ui/transactions/TransactionsActivity\_java,2

- Cursor accountsCursor = mAccountsDbAdapter.fetchAllAccounts () ;

+ Cursor accountsCursor = mAccountsDbAdapter.fetchAllRecords () ;

integration-tests/pom\_xml,3

- < artifactId > gnucash- android- app < /artifactId >

- < version > 1.1.1- SNAPSHOT < /version >

+ < artifactId > gnucash- android < /artifactId >

+ < version > 1.1.3- SNAPSHOT < /version >

integration-tests/pom\_xml,2

- < artifactId > gnucash- android- app < /artifactId >

- < version > 1.1.1- SNAPSHOT < /version >

+ < artifactId > gnucash- android < /artifactId >

+ < version > 1.1.3- SNAPSHOT < /version >

app/res/layout/dialog\_export\_xml,3

- Currency accountCurrency = Currency.getInstance ( mAccountsDbAdapter.getCurrencyCode ( mAccountId ) ) ;

+ Currency accountCurrency = Currency.getInstance ( mAccountsDbAdapter.getCurrencyCode ( mAccountUID ) ) ;

app/res/layout/dialog\_export\_xml,4

- String currencyCode = mAccountsDbAdapter.getCurrencyCode ( mAccountId ) ;

+ String currencyCode = mAccountsDbAdapter.getCurrencyCode ( mAccountUID ) ;

GnucashMobileTest/AndroidManifest\_xml,1

- android:versionCode = " 1 "

- android:versionName = " 1.0 " >

+ android:versionCode = " 2 "

+ android:versionName = " 1.1.1 " >

app/src/main/res/xml/preference\_fragment\_headers\_xml,1

- < Preference android:title = " @string/header\_manage\_books "

+ < Preference android:title = " @string/title\_manage\_books "

app/src/main/res/xml/preference\_fragment\_headers\_xml,2

- < /PreferenceScreen >

+ < /PreferenceScreen >

GnucashMobile/src/org/gnucash/android/ui/ExportDialogFragment\_java,1

-

- package org.gnucash.android.ui ;

-

- import java.io.File ;

- import java.io.FileInputStream ;

- import java.io.FileOutputStream ;

- import java.io.FileWriter ;

- import java.io.IOException ;

- import java.io.Writer ;

- import java.nio.channels.FileChannel ;

- import java.text.SimpleDateFormat ;

- import java.util.Date ;

-

- import javax.xml.parsers.DocumentBuilder ;

- import javax.xml.parsers.DocumentBuilderFactory ;

- import javax.xml.parsers.ParserConfigurationException ;

- import javax.xml.transform.OutputKeys ;

- import javax.xml.transform.Transformer ;

- import javax.xml.transform.TransformerConfigurationException ;

- import javax.xml.transform.TransformerException ;

- import javax.xml.transform.TransformerFactory ;

- import javax.xml.transform.dom.DOMSource ;

- import javax.xml.transform.stream.StreamResult ;

-

- import org.gnucash.android.R ;

- import org.gnucash.android.util.OfxFormatter ;

- import org.w3c.dom.Document ;

- import org.w3c.dom.Element ;

- import org.w3c.dom.ProcessingInstruction ;

-

- import android.app.Activity ;

- import android.content.Intent ;

- import android.net.Uri ;

- import android.os.Bundle ;

- import android.support.v4.app.DialogFragment ;

- import android.util.Log ;

- import android.view.LayoutInflater ;

- import android.view.View ;

- import android.view.ViewGroup ;

- import android.widget.ArrayAdapter ;

- import android.widget.Button ;

- import android.widget.CheckBox ;

- import android.widget.Spinner ;

- import android.widget.Toast ;

-

- public class ExportDialogFragment extends DialogFragment {

-

- Spinner mDestinationSpinner ;

- CheckBox mExportAllCheckBox ;

- Button mSaveButton ;

- Button mCancelButton ;

-

- String mFilePath ;

-

- @Override

- public View onCreateView ( LayoutInflater inflater , ViewGroup container ,

- Bundle savedInstanceState ) {

- return inflater.inflate ( R.layout.dialog\_export\_ofx , container , false ) ;

- }

-

- @Override

- public void onActivityCreated ( Bundle savedInstanceState ) {

- super.onActivityCreated ( savedInstanceState ) ;

- mFilePath = getActivity () .getExternalFilesDir ( null ) + " / " + buildExportFilename () ;

- getDialog () .setTitle ( R.string.export\_ofx ) ;

- bindViews () ;

- }

-

- private void bindViews () {

- View v = getView () ;

- mDestinationSpinner = ( Spinner ) v.findViewById ( R.id.spinner\_export\_destination ) ;

- ArrayAdapter < CharSequence > adapter = ArrayAdapter.createFromResource ( getActivity () ,

- R.array.export\_destinations , android.R.layout.simple\_spinner\_item ) ;

- adapter.setDropDownViewResource ( android.R.layout.simple\_spinner\_dropdown\_item ) ;

- mDestinationSpinner.setAdapter ( adapter ) ;

-

- mExportAllCheckBox = ( CheckBox ) v.findViewById ( R.id.checkbox\_export\_all ) ;

- mSaveButton = ( Button ) v.findViewById ( R.id.btn\_save ) ;

- mCancelButton = ( Button ) v.findViewById ( R.id.btn\_cancel ) ;

-

- mCancelButton.setOnClickListener ( new View.OnClickListener () {

-

- @Override

- public void onClick ( View v ) {

- dismiss () ;

- }

- } ) ;

-

- mSaveButton.setOnClickListener ( new View.OnClickListener () {

-

- @Override

- public void onClick ( View v ) {

- boolean exportAll = mExportAllCheckBox.isChecked () ;

- Document document = null ;

- try {

- document = exportOfx ( exportAll ) ;

- writeToExternalStorage ( document ) ;

- } catch ( Exception e ) {

- Log.e ( getTag () , e.getMessage () ) ;

- Toast.makeText ( getActivity () , R.string.error\_exporting ,

- Toast.LENGTH\_LONG ) .show () ;

- dismiss () ;

- return ;

- }

-

- int position = mDestinationSpinner.getSelectedItemPosition () ;

- switch ( position ) {

- case 0:

- shareFile ( mFilePath ) ;

-

- break ;

-

- case 1:

-

- Toast.makeText ( getActivity () ,

- " OFX file exported to:\n " + mFilePath ,

- Toast.LENGTH\_LONG ) .show () ;

- break ;

-

- default:

- break ;

- }

- dismiss () ;

- }

- } ) ;

- }

-

- private void writeToExternalStorage ( Document doc ) throws IOException {

- File file = new File ( mFilePath ) ;

-

- FileWriter writer = new FileWriter ( file ) ;

- write ( doc , writer ) ;

-

- }

-

- @Override

- public void onActivityResult ( int requestCode , int resultCode , Intent data ) {

-

- if ( resultCode == Activity.RESULT\_OK ) {

-

- File file = new File ( mFilePath ) ;

- file.delete () ;

- }

- }

-

- private void shareFile ( String path ) {

- Intent shareIntent = new Intent ( Intent.ACTION\_SEND ) ;

- shareIntent.setType ( " text/plain " ) ;

- shareIntent.putExtra ( Intent.EXTRA\_STREAM , Uri.parse ( " file:// " + path ) ) ;

- shareIntent.putExtra ( Intent.EXTRA\_SUBJECT , " Gnucash OFX export " ) ;

- SimpleDateFormat formatter = new SimpleDateFormat ( " yyyy.MM.dd HH:mm " ) ;

- shareIntent.putExtra ( Intent.EXTRA\_TEXT , " Gnucash accounts export from "

- + formatter.format ( new Date ( System.currentTimeMillis () ) ) ) ;

- startActivity ( Intent.createChooser ( shareIntent , " Sharing OFX file... " ) ) ;

- }

-

- public static void copyFile ( File src , File dst ) throws IOException

- {

-

- FileChannel inChannel = new FileInputStream ( src ) .getChannel () ;

- FileChannel outChannel = new FileOutputStream ( dst ) .getChannel () ;

- try

- {

- inChannel.transferTo ( 0 , inChannel.size () , outChannel ) ;

- }

- finally

- {

- if ( inChannel != null )

- inChannel.close () ;

- if ( outChannel != null )

- outChannel.close () ;

- }

- }

-

- public static String buildExportFilename () {

- SimpleDateFormat formatter = new SimpleDateFormat ( " yyyyMMdd\_HHmm " ) ;

- String filename = formatter.format (

- new Date ( System.currentTimeMillis () ) )

- + " \_gnucash\_all.ofx " ;

- return filename ;

- }

-

- protected Document exportOfx ( boolean exportAll ) throws ParserConfigurationException {

- DocumentBuilderFactory docFactory = DocumentBuilderFactory

- .newInstance () ;

- DocumentBuilder docBuilder = docFactory.newDocumentBuilder () ;

-

- Document document = docBuilder.newDocument () ;

- Element root = document.createElement ( " OFX " ) ;

-

- ProcessingInstruction pi = document.createProcessingInstruction ( " OFX " , " OFXHEADER = \ " 200\ " VERSION = \ " 211\ " SECURITY = \ " NONE\ " OLDFILEUID = \ " NONE\ " NEWFILEUID = \ " NONE\ " " ) ;

- document.appendChild ( pi ) ;

- document.appendChild ( root ) ;

-

- OfxFormatter exporter = new OfxFormatter ( getActivity () , exportAll ) ;

- exporter.toXml ( document , root ) ;

-

- return document ;

- }

-

- public void write ( Document document , Writer outputWriter ) {

- try {

- TransformerFactory transformerFactory = TransformerFactory

- .newInstance () ;

- Transformer transformer = transformerFactory.newTransformer () ;

- DOMSource source = new DOMSource ( document ) ;

- StreamResult result = new StreamResult ( outputWriter ) ;

-

- transformer.setOutputProperty ( " { http://xml.apache.org/xslt } indent- amount " , " 2 " ) ;

- transformer.setOutputProperty ( OutputKeys.INDENT , " yes " ) ;

-

- transformer.transform ( source , result ) ;

- } catch ( TransformerConfigurationException txconfigException ) {

- txconfigException.printStackTrace () ;

- } catch ( TransformerException tfException ) {

- tfException.printStackTrace () ;

- }

- }

- }

-

+

+ package org.gnucash.android.ui ;

+

+ import java.io.File ;

+ import java.io.FileInputStream ;

+ import java.io.FileOutputStream ;

+ import java.io.FileWriter ;

+ import java.io.IOException ;

+ import java.io.Writer ;

+ import java.nio.channels.FileChannel ;

+ import java.text.SimpleDateFormat ;

+ import java.util.Date ;

+

+ import javax.xml.parsers.DocumentBuilder ;

+ import javax.xml.parsers.DocumentBuilderFactory ;

+ import javax.xml.parsers.ParserConfigurationException ;

+ import javax.xml.transform.OutputKeys ;

+ import javax.xml.transform.Transformer ;

+ import javax.xml.transform.TransformerConfigurationException ;

+ import javax.xml.transform.TransformerException ;

+ import javax.xml.transform.TransformerFactory ;

+ import javax.xml.transform.dom.DOMSource ;

+ import javax.xml.transform.stream.StreamResult ;

+

+ import org.gnucash.android.R ;

+ import org.gnucash.android.util.OfxFormatter ;

+ import org.w3c.dom.Document ;

+ import org.w3c.dom.Element ;

+ import org.w3c.dom.ProcessingInstruction ;

+

+ import android.app.Activity ;

+ import android.content.Intent ;

+ import android.net.Uri ;

+ import android.os.Bundle ;

+ import android.support.v4.app.DialogFragment ;

+ import android.util.Log ;

+ import android.view.LayoutInflater ;

+ import android.view.View ;

+ import android.view.ViewGroup ;

+ import android.widget.ArrayAdapter ;

+ import android.widget.Button ;

+ import android.widget.CheckBox ;

+ import android.widget.Spinner ;

+ import android.widget.Toast ;

+

+ public class ExportDialogFragment extends DialogFragment {

+

+ Spinner mDestinationSpinner ;

+ CheckBox mExportAllCheckBox ;

+ Button mSaveButton ;

+ Button mCancelButton ;

+

+ String mFilePath ;

+

+ @Override

+ public View onCreateView ( LayoutInflater inflater , ViewGroup container ,

+ Bundle savedInstanceState ) {

+ return inflater.inflate ( R.layout.dialog\_export\_ofx , container , false ) ;

+ }

+

+ @Override

+ public void onActivityCreated ( Bundle savedInstanceState ) {

+ super.onActivityCreated ( savedInstanceState ) ;

+ mFilePath = getActivity () .getExternalFilesDir ( null ) + " / " + buildExportFilename () ;

+ getDialog () .setTitle ( R.string.export\_ofx ) ;

+ bindViews () ;

+ }

+

+ private void bindViews () {

+ View v = getView () ;

+ mDestinationSpinner = ( Spinner ) v.findViewById ( R.id.spinner\_export\_destination ) ;

+ ArrayAdapter < CharSequence > adapter = ArrayAdapter.createFromResource ( getActivity () ,

+ R.array.export\_destinations , android.R.layout.simple\_spinner\_item ) ;

+ adapter.setDropDownViewResource ( android.R.layout.simple\_spinner\_dropdown\_item ) ;

+ mDestinationSpinner.setAdapter ( adapter ) ;

+

+ mExportAllCheckBox = ( CheckBox ) v.findViewById ( R.id.checkbox\_export\_all ) ;

+ mSaveButton = ( Button ) v.findViewById ( R.id.btn\_save ) ;

+ mCancelButton = ( Button ) v.findViewById ( R.id.btn\_cancel ) ;

+

+ mCancelButton.setOnClickListener ( new View.OnClickListener () {

+

+ @Override

+ public void onClick ( View v ) {

+ dismiss () ;

+ }

+ } ) ;

+

+ mSaveButton.setOnClickListener ( new View.OnClickListener () {

+

+ @Override

+ public void onClick ( View v ) {

+ boolean exportAll = mExportAllCheckBox.isChecked () ;

+ Document document = null ;

+ try {

+ document = exportOfx ( exportAll ) ;

+ writeToExternalStorage ( document ) ;

+ } catch ( Exception e ) {

+ Log.e ( getTag () , e.getMessage () ) ;

+ Toast.makeText ( getActivity () , R.string.error\_exporting ,

+ Toast.LENGTH\_LONG ) .show () ;

+ dismiss () ;

+ return ;

+ }

+

+ int position = mDestinationSpinner.getSelectedItemPosition () ;

+ switch ( position ) {

+ case 0:

+ shareFile ( mFilePath ) ;

+

+ break ;

+

+ case 1:

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+ Toast.makeText ( getActivity () ,

+ " OFX file exported to:\n " + mFilePath ,

+ Toast.LENGTH\_LONG ) .show () ;

+ break ;

+

+ default:

+ break ;

+ }

+ dismiss () ;

+ }

+ } ) ;

+ }

+

+ private void writeToExternalStorage ( Document doc ) throws IOException {

+ File file = new File ( mFilePath ) ;

+

+ FileWriter writer = new FileWriter ( file ) ;

+ write ( doc , writer ) ;

+

+ }

+

+ @Override

+ public void onActivityResult ( int requestCode , int resultCode , Intent data ) {

+

+ if ( resultCode == Activity.RESULT\_OK ) {

+

+ File file = new File ( mFilePath ) ;

+ file.delete () ;

+ }

+ }

+

+ private void shareFile ( String path ) {

+ Intent shareIntent = new Intent ( Intent.ACTION\_SEND ) ;

+ shareIntent.setType ( " text/plain " ) ;

+ shareIntent.putExtra ( Intent.EXTRA\_STREAM , Uri.parse ( " file:// " + path ) ) ;

+ shareIntent.putExtra ( Intent.EXTRA\_SUBJECT , " Gnucash OFX export " ) ;

+ SimpleDateFormat formatter = new SimpleDateFormat ( " yyyy.MM.dd HH:mm " ) ;

+ shareIntent.putExtra ( Intent.EXTRA\_TEXT , " Gnucash accounts export from "

+ + formatter.format ( new Date ( System.currentTimeMillis () ) ) ) ;

+ startActivity ( Intent.createChooser ( shareIntent , " Sharing OFX file... " ) ) ;

+ }

+

+ public static void copyFile ( File src , File dst ) throws IOException

+ {

+

+ FileChannel inChannel = new FileInputStream ( src ) .getChannel () ;

+ FileChannel outChannel = new FileOutputStream ( dst ) .getChannel () ;

+ try

+ {

+ inChannel.transferTo ( 0 , inChannel.size () , outChannel ) ;

+ }

+ finally

+ {

+ if ( inChannel != null )

+ inChannel.close () ;

+ if ( outChannel != null )

+ outChannel.close () ;

+ }

+ }

+

+ public static String buildExportFilename () {

+ SimpleDateFormat formatter = new SimpleDateFormat ( " yyyyMMdd\_HHmm " ) ;

+ String filename = formatter.format (

+ new Date ( System.currentTimeMillis () ) )

+ + " \_gnucash\_all.ofx " ;

+ return filename ;

+ }

+

+ protected Document exportOfx ( boolean exportAll ) throws ParserConfigurationException {

+ DocumentBuilderFactory docFactory = DocumentBuilderFactory

+ .newInstance () ;

+ DocumentBuilder docBuilder = docFactory.newDocumentBuilder () ;

+

+ Document document = docBuilder.newDocument () ;

+ Element root = document.createElement ( " OFX " ) ;

+

+ ProcessingInstruction pi = document.createProcessingInstruction ( " OFX " , " OFXHEADER = \ " 200\ " VERSION = \ " 211\ " SECURITY = \ " NONE\ " OLDFILEUID = \ " NONE\ " NEWFILEUID = \ " NONE\ " " ) ;

+ document.appendChild ( pi ) ;

+ document.appendChild ( root ) ;

+

+ OfxFormatter exporter = new OfxFormatter ( getActivity () , exportAll ) ;

+ exporter.toXml ( document , root ) ;

+

+ return document ;

+ }

+

+ public void write ( Document document , Writer outputWriter ) {

+ try {

+ TransformerFactory transformerFactory = TransformerFactory

+ .newInstance () ;

+ Transformer transformer = transformerFactory.newTransformer () ;

+ DOMSource source = new DOMSource ( document ) ;

+ StreamResult result = new StreamResult ( outputWriter ) ;

+

+ transformer.setOutputProperty ( " { http://xml.apache.org/xslt } indent- amount " , " 2 " ) ;

+ transformer.setOutputProperty ( OutputKeys.INDENT , " yes " ) ;

+

+ transformer.transform ( source , result ) ;

+ } catch ( TransformerConfigurationException txconfigException ) {

+ txconfigException.printStackTrace () ;

+ } catch ( TransformerException tfException ) {

+ tfException.printStackTrace () ;

+ }

+ }

+ }

+

app/src/org/gnucash/android/ui/transaction/dialog/SplitEditorDialogFragment\_java,1

- Currency accountCurrency = Currency.getInstance ( mAccountsDbAdapter.getCurrencyCode ( mAccountId ) ) ;

+ Currency accountCurrency = Currency.getInstance ( mAccountsDbAdapter.getCurrencyCode ( split.getAccountUID () ) ) ;

GnucashMobile/res/values-nb/strings\_xml,1

- < string name = " title\_add\_account " > Opprette konto < /string >

- < string name = " title\_edit\_account " > Editere konto < /string >

+ < string name = " title\_add\_account " > Opprett konto < /string >

+ < string name = " title\_edit\_account " > Rediger konto < /string >

- < string name = " description\_add\_transaction\_icon " > Legge til ny konto transaksjon < /string >

- < string name = " label\_no\_accounts " > Ingen konti vise < /string >

+ < string name = " description\_add\_transaction\_icon " > Legg til ny kontotransaksjon < /string >

+ < string name = " label\_no\_accounts " > Ingen kontoer vise < /string >

GnucashMobile/res/values-nb/strings\_xml,9

- muligheten til raskt og fleksibelt logge transaksjoner som kan eksporteres til OFX ( Open Financial eXchange )

+ muligheten til raskt og fleksibelt logge transaksjoner som kan eksporteres til OFX ( Open Financial eXchange ) -

- < string name = " toast\_error\_exporting " > Kunne ikke lagre OFX fil til :\n < /string >

- < string name = " toast\_ofx\_exported\_to " > OFX fil eksportert til:\n < /string >

- < string name = " title\_export\_email " > Gnucash OFX eksportering < /string >

- < string name = " description\_export\_email " > Gnucash OFX eksportering til < /string >

+ < string name = " toast\_error\_exporting " > Kunne ikke lagre OFX- fil til :\n < /string >

+ < string name = " toast\_ofx\_exported\_to " > OFX- fil eksportert til:\n < /string >

+ < string name = " title\_export\_email " > Gnucash OFX- eksportering < /string >

+ < string name = " description\_export\_email " > Gnucash OFX- eksportering til < /string >

GnucashMobile/res/values-nb/strings\_xml,4

- < item > Norske Kroner < /item >

+ < item > Norske kroner < /item >

GnucashMobile/res/values-nb/strings\_xml,8

- < string name = " title\_build\_version " > Bygg version < /string >

+ < string name = " title\_build\_version " > Byggversion < /string >

app/src/main/java/org/gnucash/android/export/ofx/OfxExporter\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

GnucashMobile/res/menu/account\_actions\_xml,1

- android:icon = " @drawable/content\_import\_export\_holo\_dark "

+ android:icon = " @drawable/content\_import\_export\_holo\_light "

app/src/org/gnucash/android/model/Split\_java,4

- public Split ( Split sourceSplit , boolean generateUID ) {

+ public Split ( @NonNull Split sourceSplit , boolean generateUID ) {

app/src/androidTest/java/org/gnucash/android/test/ui/PieChartActivityTest\_java,2

- public class PieChartActivityTest extends ActivityInstrumentationTestCase2 < PieChartActivity > {

+ public class PieChartActivityTest extends ActivityInstrumentationTestCase2 < PieChartFragment > {

app/src/androidTest/java/org/gnucash/android/test/ui/PieChartActivityTest\_java,5

- onView ( withId ( R.id.chart\_date ) ) .check ( matches ( withText ( date.toString ( PieChartActivity.DATE\_PATTERN ) ) ) ) ;

+ onView ( withId ( R.id.chart\_date ) ) .check ( matches ( withText ( date.toString ( PieChartFragment.DATE\_PATTERN ) ) ) ) ;

app/src/androidTest/java/org/gnucash/android/test/ui/PieChartActivityTest\_java,4

- String selectedText = String.format ( PieChartActivity.SELECTED\_VALUE\_PATTERN , DINING\_EXPENSE\_ACCOUNT\_NAME , TRANSACTION\_AMOUNT , percent ) ;

+ String selectedText = String.format ( PieChartFragment.SELECTED\_VALUE\_PATTERN , DINING\_EXPENSE\_ACCOUNT\_NAME , TRANSACTION\_AMOUNT , percent ) ;

app/src/androidTest/java/org/gnucash/android/test/ui/PieChartActivityTest\_java,6

- String selectedText = String.format ( PieChartActivity.SELECTED\_VALUE\_PATTERN , GIFTS\_RECEIVED\_INCOME\_ACCOUNT\_NAME , TRANSACTION3\_AMOUNT , 100f ) ;

+ String selectedText = String.format ( PieChartFragment.SELECTED\_VALUE\_PATTERN , GIFTS\_RECEIVED\_INCOME\_ACCOUNT\_NAME , TRANSACTION3\_AMOUNT , 100f ) ;

app/src/androidTest/java/org/gnucash/android/test/ui/TransactionsActivityTest\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

- import org.gnucash.android.db.SplitsDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.SplitsDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

GnucashMobile/res/menu/account\_context\_menu\_xml,1

- android:title = " @string/delete "

+ android:title = " @string/menu\_delete "

app/src/main/res/layout/fragment\_accounts\_list\_xml,1

- android:gravity = " center "

+ android:gravity = " center "

app/src/main/java/org/gnucash/android/ui/transaction/dialog/SplitEditorDialogFragment\_java,1

- mSplitItemViewList = new ArrayList < View > () ;

+ mSplitItemViewList = new ArrayList < > () ;

app/src/main/res/drawable-hdpi/ic\_mode\_edit\_white\_24dp\_png,3

- Toolbar toolbar = ( Toolbar ) findViewById ( R.id.toolbar\_actionbar ) ;

+ Toolbar toolbar = ( Toolbar ) findViewById ( R.id.toolbar\_transaction\_info ) ;

app/src/main/res/drawable-hdpi/ic\_mode\_edit\_white\_24dp\_png,2

- mAccountUID = ( ( TransactionsActivity ) getActivity () ) .getCurrentAccountUID () ;

+ mAccountUID = ( ( FormActivity ) getActivity () ) .getCurrentAccountUID () ;

app/src/org/gnucash/android/export/xml/GncXmlHelper\_java,7

- Date date = DATE\_FORMATTER.parse ( dateString ) ;

+ Date date = TIME\_FORMATTER.parse ( dateString ) ;

app/src/org/gnucash/android/export/xml/GncXmlHelper\_java,6

- return DATE\_FORMATTER.format ( new Date ( milliseconds ) ) ;

+ return TIME\_FORMATTER.format ( new Date ( milliseconds ) ) ;

app/res/menu/global\_actions\_xml,1

- android:orderInCategory = " 4 " / >

+ android:orderInCategory = " 4 " / >

app/src/main/java/org/gnucash/android/ui/util/CustomKeyboard\_java,1

-

- package org.gnucash.android.ui.util ;

-

- import android.app.Activity ;

- import android.app.AlertDialog ;

- import android.content.DialogInterface ;

- import android.inputmethodservice.Keyboard ;

- import android.inputmethodservice.KeyboardView ;

- import android.inputmethodservice.KeyboardView.OnKeyboardActionListener ;

- import android.text.Editable ;

- import android.text.InputType ;

- import android.text.Layout ;

- import android.view.HapticFeedbackConstants ;

- import android.view.KeyEvent ;

- import android.view.MotionEvent ;

- import android.view.View ;

- import android.view.View.OnClickListener ;

- import android.view.View.OnFocusChangeListener ;

- import android.view.View.OnTouchListener ;

- import android.view.WindowManager ;

- import android.view.inputmethod.InputMethodManager ;

- import android.widget.EditText ;

-

-

- public class CustomKeyboard {

-

- private KeyboardView mKeyboardView ;

- private Activity mHostActivity ;

- private boolean hapticFeedback ;

-

- private OnKeyboardActionListener mOnKeyboardActionListener = new OnKeyboardActionListener () {

- @Override

- public void onKey ( int primaryCode , int[] keyCodes ) {

- View focusCurrent = mHostActivity.getWindow () .getCurrentFocus () ;

-

-

- EditText edittext = ( EditText ) focusCurrent ;

- Editable editable = edittext.getText () ;

- int start = edittext.getSelectionStart () ;

-

- int end = edittext.getSelectionEnd () ;

-

- if ( end > start )

- editable.delete ( start , end ) ;

-

- switch ( primaryCode ) {

- case 48:

- case 49:

- case 50:

- case 51:

- case 52:

- case 53:

- case 54:

- case 55:

- case 56:

- case 57:

- editable.insert ( start , Character.toString ( ( char ) primaryCode ) ) ;

- break ;

- case KeyEvent.KEYCODE\_BACK:

- hideCustomKeyboard () ;

- break ;

- case - 5:

- editable.delete ( start - 1 , start ) ;

- }

- }

-

- @Override

- public void onPress ( int arg0 ) {

-

- if ( hapticFeedback && arg0 != 0 )

- mKeyboardView.performHapticFeedback ( HapticFeedbackConstants.VIRTUAL\_KEY ) ;

- }

-

- @Override public void onRelease ( int primaryCode ) {}

- @Override public void onText ( CharSequence text ) {}

- @Override public void swipeLeft () {}

- @Override public void swipeRight () {}

- @Override public void swipeDown () {}

- @Override public void swipeUp () {}

- } ;

-

- public CustomKeyboard ( Activity host , int viewid , int layoutid ) {

- mHostActivity = host ;

- mKeyboardView = ( KeyboardView ) mHostActivity.findViewById ( viewid ) ;

- mKeyboardView.setKeyboard ( new Keyboard ( mHostActivity , layoutid ) ) ;

- mKeyboardView.setPreviewEnabled ( false ) ;

- mKeyboardView.setOnKeyboardActionListener ( mOnKeyboardActionListener ) ;

-

- mHostActivity.getWindow () .setSoftInputMode ( WindowManager.LayoutParams.SOFT\_INPUT\_STATE\_ALWAYS\_HIDDEN ) ;

- }

-

- public boolean isCustomKeyboardVisible () {

- return mKeyboardView.getVisibility () == View.VISIBLE ;

- }

-

- public void showCustomKeyboard ( View v ) {

- if ( v != null )

- ( ( InputMethodManager ) mHostActivity.getSystemService ( Activity.INPUT\_METHOD\_SERVICE ) ) .hideSoftInputFromWindow ( v.getWindowToken () , 0 ) ;

-

- mKeyboardView.setVisibility ( View.VISIBLE ) ;

- mKeyboardView.setEnabled ( true ) ;

- }

-

- public void hideCustomKeyboard () {

- mKeyboardView.setVisibility ( View.GONE ) ;

- mKeyboardView.setEnabled ( false ) ;

- }

-

- public void registerEditText ( int resid ) {

-

- final EditText edittext = ( EditText ) mHostActivity.findViewById ( resid ) ;

-

- edittext.setOnFocusChangeListener ( new OnFocusChangeListener () {

-

- @Override

- public void onFocusChange ( View v , boolean hasFocus ) {

- if ( hasFocus )

- showCustomKeyboard ( v ) ;

- else

- hideCustomKeyboard () ;

- }

- } ) ;

-

- edittext.setOnClickListener ( new OnClickListener () {

-

- @Override

- public void onClick ( View v ) {

- showCustomKeyboard ( v ) ;

- }

- } ) ;

-

-

- edittext.setInputType ( edittext.getInputType () | InputType.TYPE\_TEXT\_FLAG\_NO\_SUGGESTIONS ) ;

-

- OnTouchListener otl = new OnTouchListener () {

- @Override

- public boolean onTouch ( View v , MotionEvent event ) {

- if ( ! isCustomKeyboardVisible () )

- showCustomKeyboard ( v ) ;

-

-

- edittext.onTouchEvent ( event ) ;

-

- return false ;

- }

-

- } ;

-

- edittext.setOnTouchListener ( otl ) ;

- }

-

- public void enableHapticFeedback ( boolean goEnabled ) {

- mKeyboardView.setHapticFeedbackEnabled ( goEnabled ) ;

- hapticFeedback = goEnabled ;

- }

- }

+

+ package org.gnucash.android.ui.util ;

+

+ import android.app.Activity ;

+ import android.app.AlertDialog ;

+ import android.content.DialogInterface ;

+ import android.inputmethodservice.Keyboard ;

+ import android.inputmethodservice.KeyboardView ;

+ import android.inputmethodservice.KeyboardView.OnKeyboardActionListener ;

+ import android.text.Editable ;

+ import android.text.InputType ;

+ import android.text.Layout ;

+ import android.view.HapticFeedbackConstants ;

+ import android.view.KeyEvent ;

+ import android.view.MotionEvent ;

+ import android.view.View ;

+ import android.view.View.OnClickListener ;

+ import android.view.View.OnFocusChangeListener ;

+ import android.view.View.OnTouchListener ;

+ import android.view.WindowManager ;

+ import android.view.inputmethod.InputMethodManager ;

+ import android.widget.EditText ;

+

+

+ public class CustomKeyboard {

+

+ private KeyboardView mKeyboardView ;

+ private Activity mHostActivity ;

+ private boolean hapticFeedback ;

+

+ private OnKeyboardActionListener mOnKeyboardActionListener = new OnKeyboardActionListener () {

+ @Override

+ public void onKey ( int primaryCode , int[] keyCodes ) {

+ View focusCurrent = mHostActivity.getWindow () .getCurrentFocus () ;

+

+

+ EditText edittext = ( EditText ) focusCurrent ;

+ Editable editable = edittext.getText () ;

+ int start = edittext.getSelectionStart () ;

+

+ int end = edittext.getSelectionEnd () ;

+

+ if ( end > start )

+ editable.delete ( start , end ) ;

+

+ switch ( primaryCode ) {

+ case 48:

+ case 49:

+ case 50:

+ case 51:

+ case 52:

+ case 53:

+ case 54:

+ case 55:

+ case 56:

+ case 57:

+ editable.insert ( start , Character.toString ( ( char ) primaryCode ) ) ;

+ break ;

+ case KeyEvent.KEYCODE\_BACK:

+ hideCustomKeyboard () ;

+ break ;

+ case - 5:

+ editable.delete ( start - 1 , start ) ;

+ }

+ }

+

+ @Override

+ public void onPress ( int arg0 ) {

+

+ if ( hapticFeedback && arg0 != 0 )

+ mKeyboardView.performHapticFeedback ( HapticFeedbackConstants.VIRTUAL\_KEY ) ;

+ }

+

+ @Override public void onRelease ( int primaryCode ) {}

+ @Override public void onText ( CharSequence text ) {}

+ @Override public void swipeLeft () {}

+ @Override public void swipeRight () {}

+ @Override public void swipeDown () {}

+ @Override public void swipeUp () {}

+ } ;

+

+ public CustomKeyboard ( Activity host , int viewid , int layoutid ) {

+ mHostActivity = host ;

+ mKeyboardView = ( KeyboardView ) mHostActivity.findViewById ( viewid ) ;

+ mKeyboardView.setKeyboard ( new Keyboard ( mHostActivity , layoutid ) ) ;

+ mKeyboardView.setPreviewEnabled ( false ) ;

+ mKeyboardView.setOnKeyboardActionListener ( mOnKeyboardActionListener ) ;

+

+ mHostActivity.getWindow () .setSoftInputMode ( WindowManager.LayoutParams.SOFT\_INPUT\_STATE\_ALWAYS\_HIDDEN ) ;

+ }

+

+ public boolean isCustomKeyboardVisible () {

+ return mKeyboardView.getVisibility () == View.VISIBLE ;

+ }

+

+ public void showCustomKeyboard ( View v ) {

+ if ( v != null )

+ ( ( InputMethodManager ) mHostActivity.getSystemService ( Activity.INPUT\_METHOD\_SERVICE ) ) .hideSoftInputFromWindow ( v.getWindowToken () , 0 ) ;

+

+ mKeyboardView.setVisibility ( View.VISIBLE ) ;

+ mKeyboardView.setEnabled ( true ) ;

+ }

+

+ public void hideCustomKeyboard () {

+ mKeyboardView.setVisibility ( View.GONE ) ;

+ mKeyboardView.setEnabled ( false ) ;

+ }

+

+ public void registerEditText ( int resid ) {

+

+ final EditText edittext = ( EditText ) mHostActivity.findViewById ( resid ) ;

+

+ edittext.setOnFocusChangeListener ( new OnFocusChangeListener () {

+

+ @Override

+ public void onFocusChange ( View v , boolean hasFocus ) {

+ if ( hasFocus )

+ showCustomKeyboard ( v ) ;

+ else

+ hideCustomKeyboard () ;

+ }

+ } ) ;

+

+ edittext.setOnClickListener ( new OnClickListener () {

+

+ @Override

+ public void onClick ( View v ) {

+ showCustomKeyboard ( v ) ;

+ }

+ } ) ;

+

+

+ edittext.setInputType ( edittext.getInputType () | InputType.TYPE\_TEXT\_FLAG\_NO\_SUGGESTIONS ) ;

+

+ OnTouchListener otl = new OnTouchListener () {

+ @Override

+ public boolean onTouch ( View v , MotionEvent event ) {

+ if ( ! isCustomKeyboardVisible () )

+ showCustomKeyboard ( v ) ;

+

+

+ edittext.onTouchEvent ( event ) ;

+

+ return false ;

+ }

+

+ } ;

+

+ edittext.setOnTouchListener ( otl ) ;

+ }

+

+ public void enableHapticFeedback ( boolean goEnabled ) {

+ mKeyboardView.setHapticFeedbackEnabled ( goEnabled ) ;

+ hapticFeedback = goEnabled ;

+ }

+ }

app/src/main/java/org/gnucash/android/ui/report/ReportSummaryFragment\_java,1

- import org.joda.time.LocalDateTime ;

+ import org.joda.time.LocalDate ;

app/src/main/java/org/gnucash/android/ui/report/ReportSummaryFragment\_java,4

-

- GroupInterval mReportGroupInterval = GroupInterval.MONTH ;

+ private GroupInterval mReportGroupInterval = GroupInterval.MONTH ;

app/src/main/java/org/gnucash/android/export/qif/QifExporter\_java,1

- import org.gnucash.android.db.AccountsDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.AccountsDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

app/src/main/java/org/gnucash/android/ui/settings/BookManagerFragment\_java,1

- lastSyncText.setText ( PreferencesHelper.getLastExportTime () .toString () ) ;

+ lastSyncText.setText ( PreferencesHelper.getLastExportTime ( bookUID ) .toString () ) ;

app/src/main/java/org/gnucash/android/receivers/BootReceiver\_java,1

- GnuCashApplication.startScheduledEventExecutionService ( context ) ;

+ GnuCashApplication.startScheduledActionExecutionService ( context ) ;

app/src/main/res/drawable/numeric\_button\_xml,2

- android:color = " @color/abs\_\_bright\_foreground\_disabled\_holo\_light " / >

+ android:color = " @color/vpi\_\_bright\_foreground\_disabled\_holo\_light " / >

app/src/main/java/org/gnucash/android/service/SchedulerService\_java,1

- import org.gnucash.android.db.ScheduledActionDbAdapter ;

- import org.gnucash.android.db.TransactionsDbAdapter ;

+ import org.gnucash.android.db.adapter.ScheduledActionDbAdapter ;

+ import org.gnucash.android.db.adapter.TransactionsDbAdapter ;

app/res/layout/dialog\_export\_ofx\_xml,2

- android:text = " @string/option\_delete\_after\_export "

- / >

+ android:text = " @string/option\_delete\_after\_export "

+ / >

app/src/main/java/org/gnucash/android/ui/report/barchart/BarChartFragment\_java,3

- import org.gnucash.android.ui.report.barchart.BarChartFragment ;

+ import org.gnucash.android.ui.report.barchart.CashFlowBarChartFragment ;

app/res/raw-de/default\_accounts\_gnucash,9

- mSpinnerAdapter = new QualifiedAccountNameCursorAdapter ( getSupportActionBar () .getThemedContext () ,

+

+ SpinnerAdapter mSpinnerAdapter = new QualifiedAccountNameCursorAdapter ( getSupportActionBar () .getThemedContext () ,

app/src/org/gnucash/android/ui/transaction/TransactionFormFragment\_java,3

- final String[] from = new String[] { DatabaseSchema.TransactionEntry.COLUMN\_NAME } ;

+ final String[] from = new String[] { DatabaseSchema.TransactionEntry.COLUMN\_DESCRIPTION } ;

app/src/org/gnucash/android/ui/transaction/TransactionFormFragment\_java,12

- imm.hideSoftInputFromWindow ( mNameEditText.getApplicationWindowToken () , 0 ) ;

+ imm.hideSoftInputFromWindow ( mDescriptionEditText.getApplicationWindowToken () , 0 ) ;

app/src/org/gnucash/android/ui/transaction/TransactionFormFragment\_java,5

- mNameEditText.setOnItemClickListener ( new AdapterView.OnItemClickListener () {

+ mDescriptionEditText.setOnItemClickListener ( new AdapterView.OnItemClickListener () {

- if ( ! GnuCashApplication.isDoubleEntryEnabled ( true ) ) {

+ if ( ! GnuCashApplication.isDoubleEntryEnabled ( true ) ) {

app/src/org/gnucash/android/ui/transaction/TransactionFormFragment\_java,4

- final int colIndex = cursor.getColumnIndexOrThrow ( DatabaseSchema.TransactionEntry.COLUMN\_NAME ) ;

+ final int colIndex = cursor.getColumnIndexOrThrow ( DatabaseSchema.TransactionEntry.COLUMN\_DESCRIPTION ) ;

app/src/org/gnucash/android/importer/GncXmlImporter\_java,1

- GncXmlHandler handler = new GncXmlHandler ( db ) ;

+ GncXmlHandler handler = new GncXmlHandler () ;

app/src/test/java/org/gnucash/android/test/unit/model/AccountTest\_java,1

- import org.gnucash.android.test.unit.util.GnucashTestRunner ;

- import org.gnucash.android.test.unit.util.ShadowCrashlytics ;

- import org.gnucash.android.test.unit.util.ShadowUserVoice ;

+ import org.gnucash.android.test.unit.testutil.GnucashTestRunner ;

+ import org.gnucash.android.test.unit.testutil.ShadowCrashlytics ;

+ import org.gnucash.android.test.unit.testutil.ShadowUserVoice ;