Account Related:

Forgot username/password:

Send email to user containing one-time key that allows access

To change password screens

$User->getUserProfile($userName)

$User->editUserProfile($user, $editInfo)

$Message->sendEmail($userEmail)

Sign Up: (

Register user and company, capturing all appropriate profile info. This

Information can be captured from Google or Facebook using an oath-based

Register using Facebook/Google info. Subsequent screens capture shop profile.

Prospective users will be asked to identify company affiliation by selecting

from a list of already-registered companies, or entering a new company profile.

$User->addUser($info)

$User->addUserToCompany($user, $company,$companyRole)

$User->oathFacebookRegInfo($facebookId)

$User->oathGoogRegInfo($googleId)

Get approval/rejection:

Registration requests come to admin as a message of a particular type.

$User->changeUserRole($user, $role)

$Message->getMessagesByType($userId, $filter)

$User->removeUser($userId)

Ask questions about registration:

Questions come as messages of a particular type.

$Message->addNewMessage($msginfo)

Create/edit/store profile:

User enters profile information for store. Creator of store profile becomes

store-admin until that is changed. Store-admin ok’s registration requests

from prospective users identifying affiliation with that store, and sets

company role for these people.

$User->getUserProfile($userName)

$User->addUser($info)

$User->editUserProfile($user, $editInfo)

$User->setCompanyRole($user, $company, $companyRole)

Preview Store Profile:

Shows all the info the system has on that store

Company->getCompanyByName($companyName) - change: return obj not id

Add/Edit payment Info:

Sets a company’s payment pref’s and identifies payment processor.

BillInfo->addBillInfo($companyId, $newInfo)

BillInfo->editBillInfo($billInfoId, $editInfo)

BillInfo->removeBillInfo($billInfoRecordId)

BillInfo->getBillInfoById($billInfoId)

BillInfo->getBillInfoForCompany($companyId)

(BillInfo may need chaging when cc processing is determined)

Browse:

View Shop Profile:

View all info on a shop including all their catalogs w drill-down

To show catalog contents.

$Company->getCompanyByName($companyName)

$Collections->getCollectionsByCompanyId($companyId)

Add/Post Shop to favorites:

Shop is added to user’s favorites list.

$User->addShopToFavorites($companyId)

Share post/shop (need discussion):

$Message->sendMessageToGroup($msgInfo, $groupId)

Message Shop:

Message is created that is sent to shop’s admin role person.

$User->getUserInCompanyRole($user, $copmpanyRole)

$Message->addNewMessage($msginfo)

Take Action:

Shop WholesaleCatalog:

Catalogs identified as wholesale catalog type are shown for a company.

$Collection->getCollectionByCompanyAndType($company, $collectionType)

Request Sample:

Message is sent to shop-admin requesting sample.

$Message->addNewMessage($msginfo)

Buy One Sample/Buy Product(s):

Order is placed for sample product(s). Same as any order except it is q1.

$Orders->startNewOrder($company, $user)

$Orders->addProductToOrder($orderId, $productToAdd, $optionsSelected, $thisShipTypeId, $quantity)

$Orders->getIncompleteOrders($companyId)

$Orders->placeOrder($orderId)

$Orders->selectPayment($orderId) - Need to finalize payment processing

$Orders->selectShippingOption($orderId)

Browse/Select Product:

Products are displayed with option to add product to order. Option to see

similar(competing) products from same categories.

$Product->getAllSellableProductsByCategory($category)

$Product->getCompanyProducts($companyId)

$Product->getSimilarProducts($productId)

Adjust color/type:

During the adding a product to an order – can set options (color size etc)

$Options->getDefaultOptionsForProducttype($productType)

$Orders->addProductToOrder($orderId, $productToAdd, $optionsSelected, $thisShipTypeId, $quantity)

Confirm Order: (what does “add optional charge mean?)

Confirm the placement of the order, and forward it to seller.

$Orders->placeOrder($orderId)

$Orders->selectPayment($orderId) - Need to finalize payment processing

$Orders->selectShippingOption($orderId)

Add note to seller:

Seller is sent a message referencing the order.

$Message->addNewMessage($msginfo)

Save Order In Progress:

Order status set to in-progress.

$Orders->startNewOrder($company, $user)

$Orders->addProductToOrder($orderId, $productToAdd, $optionsSelected, $thisShipTypeId, $quantity)

$Orders->getIncompleteOrders($companyId)

Cancel Order:

Order status set to canceled. Order not deleted and can be revived later

$Orders->setStatusCanceled($orderId)

Return To Saved Order:

List to canceled orders is displayed. Order can be set to status in-progress

$Orders->getLineItems($orderId)

$Orders->getOrderAll($orderId)

Orders-getCanceledOrders($companyId)

Find Joonley Sellers:

List of companies either by location or product types shown.

$Company->findCompaniesByLocation($location)

$Company->findCompaniesByProductTypes($productTypes)

Find/Launch Seller Catalogs:

Get catalogs that are open for sale of a specific company

$Collections->getCollectionsByCompanyId($companyId)

Place digital order at show:

Capture barcode from camera. Include in favorites list or order.

$Company->findCompanyFromCameraBarcode()

$Product->findProductFromCameraBarcode()

$Orders->addProductFromCameraBarcode()

$Orders->placeOrder($orderId)

Save shop to favorites:

Add shop to favorites list.

$Company->saveCompanyToFavorites($companyId)

$Orders->duplicateOrder($orderId)

Manage Orders:

View existing orders with specified status.

View Open Orders:

View Confirmed Orders:

View Shipped Orders:

View Declined Orders:

View Saved Orders:

$Orders->getOrdersByStatus($companyId, $status)

$Orders->getCompanyOrdersByStatus($company, $status)

$Orders->getUserOrdersByStatus($user, $status)

$Orders->getLineItems($orderId)

See Total Value of Orders:

Total up the value of all pending orders.

Orders->getValueOfSavedOrders($companyId)

See Total Saved:

See all the orders that have been saved but not yet finalized.

$Orders->getOrderTotals($company, $status)

Add product to possibly purchase.

Add product to a list of products buyer is thinking or odering.

Order->addToFavorite(productId, $quantity)

Get value of possible purchase list

Total up value of possible purchases.

Order->getValueOfFavorites($company)

Open Orders – message seller:

Send message to seller of open order.

$Message->addNewMessage($msginfo)

Open Order – cancel an open order:

Cancel an open order and message the seller.

$orders->setStatusCanceled($orderId)

$Orders->removeStartingOrder($orderId)

$Message->addNewMessage($msginfo)

Open Order – edit open order:

Edit the details of an order before it’s placed – options,items, terms etc.

$Orders->editOrder($orderId)

$Orders->removeLineItem($OrderId, $lineItemId)

$Orders->addProductToOrder($orderId, $productToAdd, $optionsSelected, $thisShipTypeId, $quantity)

Confirmed Orders – message seller:

Shipped Orders – message seller:

Declined Orders- message seller:

Message the seller.

$Message->addNewMessage($msginfo)

Confirmed Orders – re-order:

Shipped Orders – re-order:

Duplicate order offering opportunity to edit quantity, options etc.

$Orders->getOrderWithLineItems($orderId)

$Orders->duplicateOrder($orderId)

$Orders->placeOrder($orderId)

Declined Orders – edit order:

Edit a declined order, set items, terms, options etc. Can re-activate it.

$Orders->editOrder($orderId)

$Orders->removeLineItem($OrderId, $lineItemId)

$Orders->addProductToOrder($orderId, $productToAdd, $optionsSelected, $thisShipTypeId, $quantity)

Declined Orders – clear order:

Remove a declined order from system entirely.

$Orders->removeOrder($orderId)

Saved Orders – cancel order:

Set status of in-progress order to canceled.

$orders->setStatusCanceled($orderId)

$Orders->removeStartingOrder($orderId)

Saved Orders – resume order:

Place an in-progress order

$Orders->getOrderWithLineItems($orderId)

$Orders->placeOrder($orderId)

Search:

Free Form Search:

A key-word search based on product descriptions with type filtering.

$Product->searchProductDescriptions($targetText, $filters)

Product Category Search:

A step-wise exploration of category tree.

$NestedCategory->findChildNodes($parentNodeName)

Shop Location Search:

A search or shops based on their location determined by zipcode

$Company->findCompaniesNearZip($zipcode)

Shop LocationGroup Search:

A search based on self-assigned location codes ex: east coast, New Yord etc.

$Company->findCompanyByLocationCode($locationCode)

Messageing: (needs more discussion)

Initiate nessage with shop:

Send message to shop-admin

$Message->addNewMessage($msginfo)

Access message center:

See all your current messages.

$Message->getMessagesToUser($user) – need to add filter

See previous message:

Page to previous message.

$Message->getPreviousMessage($messageId)

See next message:

Page to next message.

$Message->getNextMessage($messageId)

Get All Message Headers in thread

See all message headers in current thread.

$Message->getMessagesInThread($messageId)

Add message to thread:

Add a reply message to current thread.

$Message->addMessageToThread($messageId, $msginfo)

Add picture to message:

Upload or add camera picture to current message

$Message->uploadPicture()

$Message->addPicture($messageId)

Mark thread:

Mark a thread read or unread. Will revert to read if new reply added.

$Message->markThread(readUnreadStatus)

Delete Message:

Remove a message from the system entirely.

$Message->removeMessage($messageId)

Specials:

View Current Specials:

View products that are contained within collections of type specials.

$Product->getProductsInCollection($collectionId, $categoryFilter)

$product->getProductsInSpecialCollections($categoryFilter)

Get Participating Stores:

View a list of stores with products in specials collections.

$Collections->getCompaniesHavingCollection($collectionId)

Add Product to Special:

Add a product to a collection types as a specials collection

$Collections->addProductToCollection($productId, $collectionId, $containedAsId)

Set end date for special:

Set an expire date for a specials collection;

$Collections->setEndDate($collectionId)

Remove product from special:

Remove a product from a specials collection.

$Collections->removeProduct($productId, $collectionId)

Get sales figures on special:

Get sales data from products in a specials collection

Collections->getSalesFromCollection($collectionId, $filter)

Menus/Other:

View News and Updates:

View messages typed as news or promotional annopuncements

Messages->getFilteredMessagesToUser($userId, $filter)

User Settings Menu:

Set user-controllable settings (needs discussion)

User->editUserProfile($user, $editInfo)

Admin - System Settings (product categories):

Settings that can be made by admin – categories etc.

NestedCategory->addCategory($name, $description, $parentNodeName)

NesterCategory->deleteCategory($nodeName)

Super- Admin – Event Types:

Event types – only super-admin (developer) will be interested in this.

Events->addEventType($eventType)

Events->removeEventType($eventType)

Super-Admin – Permissions

Set permissions for various roles.

Roles->addRole($roleName, $permissions)

View Privacy Policy:

Display privacy policies (may vary by country)

User->viewPrivacyPolicy($locationId)

Access Help:

Context-sensitive help.

$User->help($pageLocation, $locationId)

Catalogs:

Preview Catalog:

Show preview of catalog

$Collections-> getCollectionsByCompanyId($companyId)

Create Catalog:

Set up a new (empty) catalog

$Collections-> addCollection($collectionName, $collectionSlug,$collectionDescription,$collectionType,$collectionStatus, $collectionCompany)

Edit Catalog:

Edit existing catalog, showing, adding and deleting products

$Collections-> editCollection($collectionId,$collectionName, $collectionSlug,$collectionDescription,$collectionType,$collectionStatus, $collectionCompanys)

$Collections->removeProductFromCollection(($productId, $collectionId)

$Collections-> addProductToCollection($productId, $collectionId, $containedAsId)

Publish Catalog:

Set status of catalog to make it visible to prospective buyers.

$Collections->setCollectionStatus($status)

Add Product to Catalog:

Add a product to an existing catalog

$Collections-> addProductToCollection($productId, $collectionId, $containedAsId)

Remove Product From Catalog:

Remove the association of aproduct with an existing catalog.

$Collections->removeProductFromCollection(($productId, $collectionId)

Products:

Add New Product:

Enter a new product using defaults (options, terms)associated with a product type.

Product->addProductUsingDefaults($productInfo)

Options->linkDefaultOptionsToProduct($productType, $productId)

Terms-> addDefaultTermsToProduct($companyId, $productId)

Set Default Options for Product Type:

Establish the default set of options and terms for a product type.

Options->addDefaultOption($option, $productType)

Associate Default Options with Product:

Associate a set of default options with a product (color, size etc).

Options->linkDefaultOptionsToProduct($productType, $productId)

Remove Product:

Remove a product entirely

Product-> removeProduct($productId)