

# Application Specifications

Application name: *Trade Manager*

Version: 1.0 (currently runs on Sandbox only)

Developer: *Sunfan Wu*

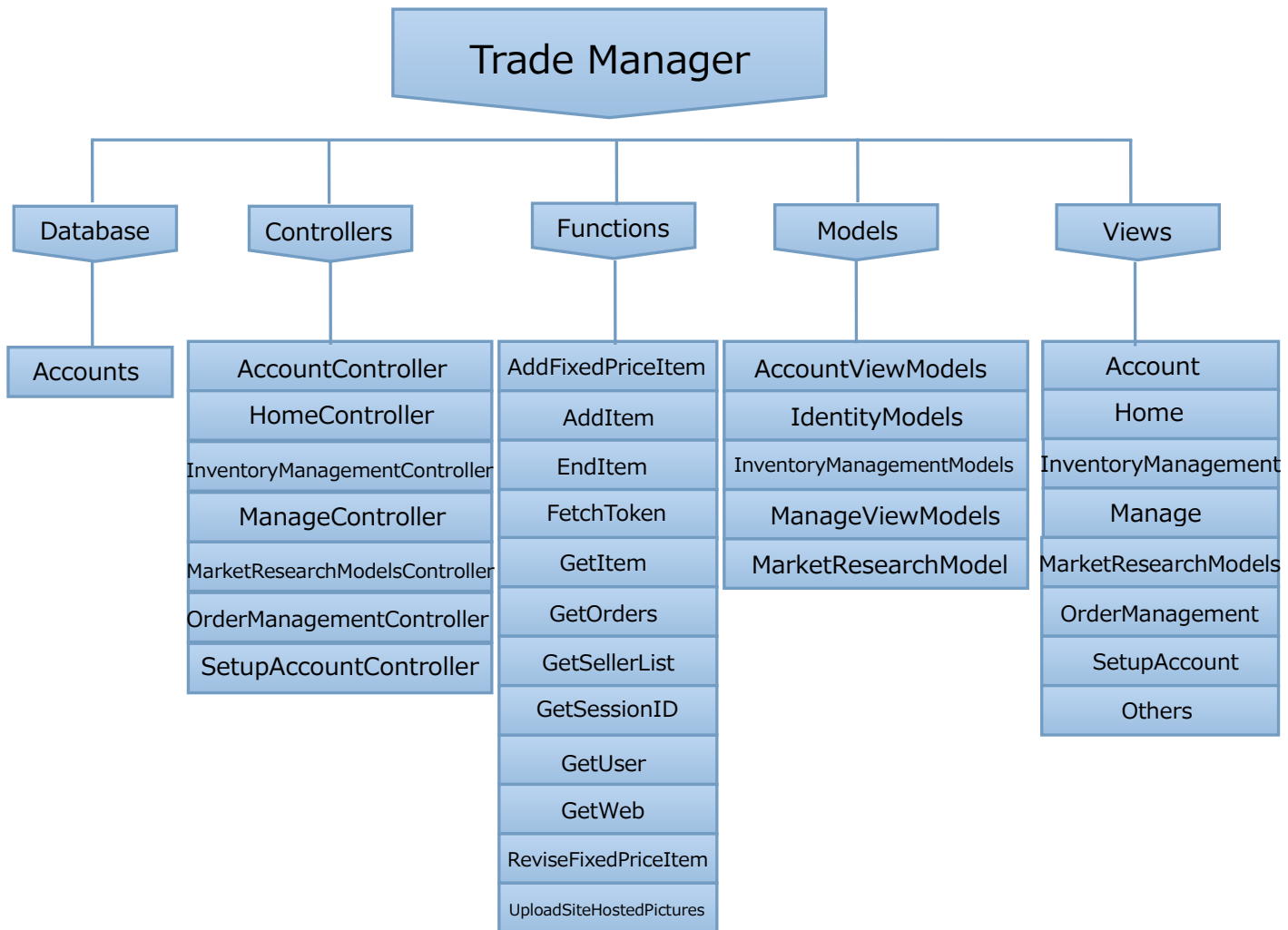
Release Date: 7/2/2015

## Generalization

This application mainly focuses on facilitating online store managements. It can currently be used on Ebay only, and is expected to be available for other sites in the future updates. The application achieves the following functions:

1. Inventory Management module: A user can build his own online inventory, items on which could be easily added, adjusted, and ended on Ebay.
  - a) Communication Module - Contacting the online store, and change the selling status dynamically.
  - b) Adjust the availability of online listing according to the specific SKU of each item.
  - c) Store the inventory information - Location of the stock, time the item is added, stock quantity, etc.
  - d) Price changing: A user can manually change the listing price of an item according to the Market Research Module results.
2. Order Management Module: This module is mainly used to manage the online selling status.
  - a) Transaction: It could check the details of each transaction made by customers. Including transaction price, buyer ID, checkout status, etc.
  - b) Selling Label: It could automatically list the selling label details of transactions made 24 hours before yesterday 10:00pm.
3. Market Research: This mainly focuses on market management/ market research.
  - a) Search the overall selling status of an item. For example: the information of recently top selling products.
  - b) Find out the price of similar products, and use the Inventory Management Module to change the price to optimize profit.

## Software structure:



## Database

### 1. Accounts.mdf:

**AspNetUsers:** contains basic user information like email, password, etc.

**InventoryManagementModels:** contains Inventories of different users.

**MarketResearchModels:** contains various entries of market researches.

## **Controllers**

### **1. AccountController.cs:**

Controls basic login and account functions.

### **2. HomeController.cs:**

1. **Index():** Login page.

### **3. InventoryManagementController.cs:**

1. **Index():** Shows all inventories in user account.
2. **Details():** Shows specific details of an item.
3. **Create():** Add a new item to inventories.
4. **Edit():** Edit existing items, or edit online seller lists(on Ebay).  
Uses: **ReviseFixedPriceItem()**
5. **Delete():** Delete an item.
6. **AddToList():** Add an Item to online seller lists.  
Uses: **AddFixedPriceItem()**
7. **SellingList():** Shows all items on online seller lists.  
Uses: **GetSellerList()**
8. **EndItem():** End an item on online seller lists.  
Uses: **EndItem()**

### **4. ManageController:**

basic account management.

### **5. MarketResearchModelsController:**

1. **index():** Shows all market researches.

2. **Refresh()**: Update an existing market research.
3. **Create()**: Add a new market research entry.  
uses: **GetWeb()**
4. **Edit()**: Edit an existing market research.  
uses: **GetWeb()**
5. **Delete()**: Delete an entry.

## **6. OrderManagementController.cs:**

1. **Index()**: Shows all online transactions.  
uses: **GetOrders()**
2. **ShippingLabels()**: Shows all shipping details of transactions within 24-hours.  
uses: **GetOrders()**

## **7. SetupAccountcontroller.cs:**

1. **Index()**: Shows Instructions to get an Ebay Token,
2. **SessionIDPage()**: Gets a temporary Ebay Session ID.  
uses: **GetSessionID()**
3. **FetchTokenPage()**: Gets Ebay Token.  
uses: **FetchToken()**

## **Functions**

### **1. AddFixedPriceItem.cs:**

Defines and lists a new fixed-price listing.

### **2. AddItem.cs:**

Defines a single new item and lists it on a specified eBay site.

### **3. EndItem.cs:**

Ends the specified item listing before the date and time at which it would normally end per the listing duration.

#### **4. FetchToken.cs:**

Retrieves an authentication token for a user.

#### **5. GetItem.cs:**

Returns item data such as title, description, price information, seller information, and so on, for the specified **ItemID**.

#### **6. GetOrders.cs:**

Retrieves the orders for which the authenticated user is a participant, either as the buyer or the seller.

#### **7. GetSellerList.cs:**

Returns a list of the items posted by the authenticated user, including the related item data.

#### **8. GetSessionID.cs:**

Retrieves a session ID that identifies a user and your application when you make a FetchToken request.

#### **9. GetUser.cs:**

Retrieves data pertaining to a single eBay user.

#### **10. GetWeb.cs:**

Retrieves information on a specific Webpage.

#### **11. ReviseFixedPriceItem.cs**

Enables a seller to change the properties of a currently active fixed-price listing (including multi-variation listings).

#### **12. UploadSiteHostedPictures.cs**

Uploads a picture to the eBay Picture Service and returns a URL of the picture.

### **Models**

#### **1. AccountViewModels.cs:**

Basic Login information like Email, Password, etc.

#### **2. IdentityModels.cs:**

**1. Token:** Stores Ebay Token for an User.

### **3. InventoryManagementModels.cs:**

1. **ID:** ID for each Item.
2. **SKU:** SKU of an item.
3. **Title:** Title of an item.
4. **LocationStock:** Location of an item on stock.
5. **Quantity:** Item quantity.
6. **StockTime:** Time when an item is stocked.
7. **Description:** Description for an item.
8. **CategoryID:** Ebay Category ID
9. **StartPrice:** Price of an Item.
10. **ListingDuration:** How long each item will be listed on Ebay.
11. **UserID:** Specifies which user the item belongs to.

### **4. ManageViewModels.cs:**

Basic user information.

### **5. MarketResearchModel.cs:**

1. **ID:** ID for each market research entry.
2. **Keyword:** Searching keyword.
3. **PriceFrom:** Lower price limit of an search.
4. **PriceTo:** Higher price limit of an search.
5. **AveragePrice:** Average price of all items searched.(number of all items <= 50)
6. **UserID:** Specifies which user the market research belongs to.

## **Views**

### **1. Account**

1. \_ExternalLoginsListPartial.cshtml
2. ConfirmEmail.cshtml
3. ExternalLoginConfirmation.cshtml
4. ExternalLoginFailure.cshtml
5. ForgotPassword.cshtml
6. ForgotPasswordConfirmation.cshtml
7. Login.cshtml
8. Register.cshtml

9. ResetPassword.cshtml
10. ResetPasswordConfirmation.cshtml
11. SendCode.cshtml
12. VerifyCode.cshtml

## **2. Home**

1. Index.cshtml

## **3. InventoryManagement**

1. AddToList.cshtml
2. Create.cshtml
3. Delete.cshtml
4. Details.cshtml
5. Edit.cshtml
6. EndItem.cshtml
7. Index.cshtml
8. SellingList.cshtml

## **4. Manage**

1. AddPhoneNumber.cshtml
2. ChangePassword.cshtml
3. Index.cshtml
4. ManageLogins.cshtml
5. Setpassword.cshtml
6. VerifyPhoneNumber.cshtml

## **5. MarketResearchModels**

1. Create.cshtml
2. Delete.cshtml
3. Edit.cshtml
4. index.cshtml
5. Refresh.cshtml

## **6. OrderManagement**

1. Index.cshtml
2. ShippingLabels.cshtml

## **7. SetupAccount**

1. FetchTokenPage.cshtml
2. Index.cshtml
3. SessionIDPage.cshtml

## **8. Other**

1. Shared
  - a) \_Layout.cshtml
  - b) \_LoginPartial.cshtml
  - c) Error.cshtml
  - d) Lockout.cshtml
2. \_Viewstart.cshtml