SimpleStore Site Requirements

# Problem Statement

An online sales presence increases a retailer’s market exponentially. Many small-business products are so niche as to make any other sales approach economically unfeasible. Selling things online, however, is a technically challenging proposition – there is hardware to procure and maintain, software to write, credit card transactions to process, and customers to service. Thus, many small retailers have made eBay their online selling venue. eBay gives users a trusted platform to sell products – it takes care of the complicated search and payment aspects of the process. However, using the eBay platform requires sellers to forego having their own product catalog website, with the associated benefits of increased marketing, better product presentation, up-sell and cross-sell opportunities, and customer support.

# Project Goal

The SimpleStore project is designed to address this shortcoming. Sellers will create and maintain a catalog of products, along with any other marketing material they need to publish on their site. However, the product catalog will integrate with eBay, allowing users to automatically insert their products as eBay listings through eBay’s API. The seller’s site will also allow customers to search for their products, once again leveraging eBay’s API. This allows the site to “own” the retail experience, while still using the eBay platform for its payment processing and fulfillment tools. Any listed products will also appear on eBay itself, increasing visibility. The site will create professional-looking template html for the eBay listing, and allow for linking between eBay and the actual SimpleStore site.

# Domain Driven Design Artifacts

Domain Driven Design (DDD) is more of a set of structures and practices used to help make complex software development map more closely to the real-world domain it is attempting to mimic or model. This particular project is not really complex enough to warrant the use of DDD, but I wanted to work through some of the patterns and tasks as a learning experience.

## Bounded Contexts

Bounded Contexts represent different models of the real world that must interact. For example, within an organization a employee might represent a work resource to his manager, a consumer of benefits to the HR department, and a creator of trouble tickets to the IT department. Each of these groups has a different model that they place the employee into. When these models meet, and especially when they need to interact, confusion can occur. Thus DDD seeks to proactively identify these separate contexts, and pay special attention to where their edges meet.

There are only two contexts within the SimpleStore project: The SimpleStore context, and the eBay context. There is significant overlap between the entities and concepts of the two sites, for example our concept of what a Product or a Category represent are similar but different; but this potential source of confusion is mitigated by very light integration necessary into the eBay context; we need only deal with a few entities on the eBay side.

## Entities

Entities represent the major classes and objects which the domain must manipulate. Entities are distinguished from other classes in that they must have a Unique identity. For example, as a person I am unique – there is no one else like me. However, as a Lead Developer I am not unique. There are other employees in my organization with that job title. The name “Patrick Brossard” is also not unique; surprisingly there are quite a few others!

|  |  |
| --- | --- |
| Entity | Description |
| Site | Each separate Store created and managed by a Seller |
| Seller | The eBay user managing one or more Sites |
| Page | A distinct Page of a Site |
| NewsArticle | Basically a blog post for the homepages of the Sites |
| Product | A product displayed on a Site. The product may or may not map to an actual physical product. |
| Listing | An instance of a product which is currently listed for sale on eBay. A Listing generally maps to an actual physical product. |

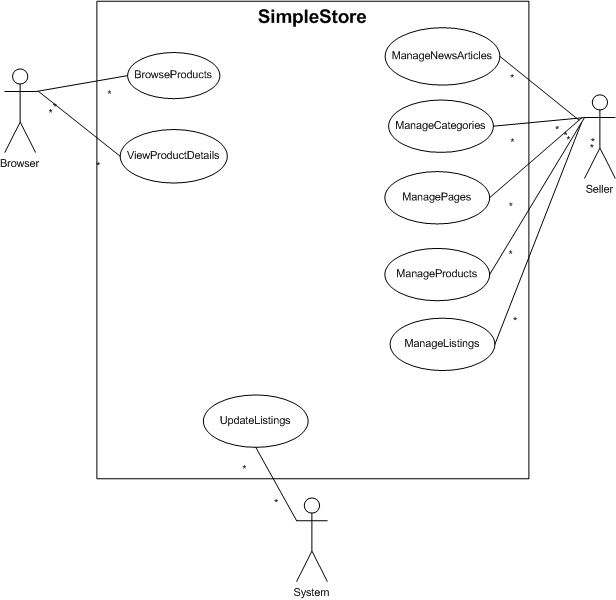
## Value Objects

As opposed to Entities, value objects are classes within the domain which do not have a unique identity. Despite this, they are often import parts of the object model, and commonly form constituent parts of the Entities. For convenience we do not consider the built-in framework objects in this list, objects such as DateTime or Currency.

|  |  |
| --- | --- |
| Value Object | Description |
| Category | A node in a Site’s category hierarchy |
| Html | A chunk of Html markup used to render content on a Site |
| ProductState | One of the states a product can be in while on a Site |
| Image | One of many images used on each Site |

# Use Case Diagram

Standard UML diagrams are still used alongside other design methodologies such as Domain Driven Design. Though they represent a form of “front-loading” design that many agile methodologies seek to minimize, it is important to have somewhere to start, and an initial prototype target to begin iterating against.



# Use Case Details

## Browser Use Cases

|  |  |
| --- | --- |
| UC-B-01: | Browse Products |
| Priority: | Essential |
| Use Frequency: | Often |
| Prerequisites: | None |
| Success Scenario: | 1. Browser visits site 2. Browser selects product category    1. Optionally, Browser enters a search term 3. Browser views product listings    1. Optionally, Browser sorts listings by Current Price, Product Title, or Popularity (default)    2. Optionally, Browser pages through product listings if more than 20 (default) are returned |
| Notes: | Category browsing and searching are basically identical but for the “breadcrumb” string showing how the user arrived at this view |

|  |  |
| --- | --- |
| UC-B-02: | View Products Details |
| Priority: | Essential |
| Use Frequency: | Often |
| Prerequisites: | UC-B-01 Browse Products |
| Success Scenario: | 1. Browser selects a product from a result list or link 2. Browser views product content    1. Optionally, Browser may select alternative images to view full size    2. Optionally, Browser may select a “Get it on eBay” link to navigate to eBay’s listing of the product. A new window is opened for this action. |
| Notes: | Product details view price, bids, and closing times do not automatically update like they do on eBay’s site. |

## Seller Use Cases

|  |  |
| --- | --- |
| UC-S-01: | Manage News Articles |
| Priority: | Desired |
| Use Frequency: | Sometimes |
| Prerequisites: | None |
| Success Scenario: | 1. Seller selects the Manage News navigation item 2. Seller views recent News listing    1. Optionally, Seller may enter a new Title and Abstract to create a new News Article.    2. Optionally, Seller may select the Edit link for an existing News Article. This places the Title and Abstract content into textboxes so the Seller may make updates. Clicking Save applies the updates, selecting Cancel… cancels them. |
| Notes: | There is no ability to override the date. The date entered is the date the News Article is associated with. Editing a News Article does not affect the date. |

|  |  |
| --- | --- |
| UC-S-02: | Manage Categories |
| Priority: | Desired |
| Use Frequency: | Sometimes |
| Prerequisites: | None |
| Success Scenario: | 1. Seller selects the Manage Categories navigation item 2. Seller views current Category Hierarchy 3. Seller may add a new Category by entering a new Category name in the New Category textbox and clicking Add. 4. Seller may rename any Category by editing its name in the associated textbox and clicking Save. 5. Seller may reorder Categories by moving them up and down in the hierarchy by selecting the ˅ and ^ buttons. 6. Seller may change the indentation level of Categories by selecting the < and > buttons. |
| Notes: | Category names need not be unique – there could be Accessories categories underneath multiple other nodes, for instance. |

|  |  |
| --- | --- |
| UC-S-03: | Manage Pages |
| Priority: | Desired |
| Use Frequency: | Rarely |
| Prerequisites: | None |
| Success Scenario: | 1. Seller selects the Manage Content Pages navigation item 2. Seller views current list of Pages, sorted by type    1. Seller may add a new Custom Content page by clicking Add New Page.    2. Seller may edit an existing Page by clicking the Page title. 3. Seller continues on to Page Details 4. Seller views Images currently associated with this Page    1. Seller may add a new Image by using the Browse/Upload controls    2. Seller may remove existing Images by using the Delete link 5. Seller may alter the Html content in the central text area control 6. Seller may preview changes before saving them. Clicking the Preview button populates the lower preview section with… a preview of the content created in the Html text area. |
| Notes: | Associated Images cannot be edited; you must delete and re-upload. |

|  |  |
| --- | --- |
| UC-S-04: | Manage Products |
| Priority: | Essential |
| Use Frequency: | Often |
| Prerequisites: | None |
| Success Scenario: | 1. Seller selects the Manage Products navigation item    1. Optionally, Seller may also select one of the Category Hierarchy links under the Manage Products navigation item. 2. Seller views the list of Products    1. Seller may elect to sort the Product list by Price, Title (default), or Popularity    2. Seller may page through the Product list if there are more than 20 (default) products returned. 3. Seller may proceed to Edit a Product by clicking on the Product title of the Edit link    1. Alternatively, Seller may proceed to Add a new Product by clicking the Add New Product link in the table header 4. Seller views the Product Details screen. 5. Seller may update the Product title in the Title texbox 6. Seller views any listings of this product currently active on eBay 7. Seller may remove or add new images associated with this Product 8. Seller may update the Short Description of the Product 9. Seller may associate a Long Description Content Page with the Product 10. Alternatively, the Seller may create a new Content Page for the Product’s long description by selecting the New Content Page link. |
| Notes: | Associated Images cannot be edited; you must delete and re-upload. |

|  |  |
| --- | --- |
| UC-S-05: | Manage Listings |
| Priority: | Essential |
| Use Frequency: | Often |
| Prerequisites: | None |
| Success Scenario: | 1. Seller selects the Manage Listings navigation item 2. Seller views the list of Listings    1. Seller may elect to sort the Listings by Price, Title, or Ending Soonest (default)    2. Seller may page through the Listings list if there are more than 20 (default) products returned. 3. Seller may proceed to view the List on eBay by clicking the View on Ebay link 4. Seller may proceed to view the Product Details by clicking on the View Product link    1. From Product Details, Seller may then create a new Listing for that Product    2. Seller completes the Add Listing process at eBay, with Product information, images, and descriptions prepopulated from the Product Details data. |
| Notes: | Add Listing process has not been completely scoped |

## System Use Cases

|  |  |
| --- | --- |
| UC-SYS-01: | Update Listings |
| Priority: | Essential |
| Use Frequency: | Always |
| Prerequisites: | None |
| Success Scenario: | 1. System Listing Update is triggered by system timer (5 minute default) 2. System connects to eBay API for each Seller account 3. System updates list of Active listings 4. System updates current price, number of bids, and time left for each Active listing |
| Notes: | Sellers can make changes to listings completely outside of the SimpleStore site. This process attempts to keep things up to date and in sync. |

# References

Domain Driven Design *Quickly*  
<http://www.infoq.com/minibooks/domain-driven-design-quickly>