



# *Wine Information Network*

# Use Cases

## Authors

Shannon Harris – Project Manager

Eliah Overbey – Senior Systems Analyst

Michael Nguyen - Software Development Lead

Andrew Hwang - Database Specialist

Brian La - Quality Assurance Lead

David Smith - Software Architect

Rex West - Subject Matter Expert

Derek Chan – Algorithm Specialist

Kyle Gullicksen - Algorithm Specialist

Edward Wang – User Interface Specialist

Compiled by: Eliah Overbey

Compilation Date: 4/20/2013

Last Updated: 6/8/2013

## Table of Contents

Information Tables .....	3
IS (Inventory System) Use Cases .....	4-17
IS-1: Add a Wine .....	4
IS-2: Add a Storage Location .....	5
IS-3: Edit a Storage Location .....	6
IS-4: Editing Entries.....	7
IS-5: Importing/Exporting Inventory .....	8
IS-6: Removing Entries .....	9
IS-7: Viewing Statistics.....	10
IS-8: Filter Inventory .....	11
IS-9: Archive/History of Inventory .....	12
IS-10: Search Inventory.....	13
IS-11: Wine Wishlist.....	14
IS-12: Inventory Security .....	15
IS-13: Rate and Review .....	16
IS-14: Remove an Inventory Location .....	17
RS (Recommender System) Use Cases .....	18-26
RS-1: View Recommendations .....	18
RS-2: Shuffle Channels.....	19
RS-3: Buy Wine.....	20
RS-4: Buddy Recommendations .....	21
RS-5: Create a Shelf .....	22
RS-6: Remove a Shelf.....	23
RS-7: Navigate Shelves.....	24
RS-8: Recommend Random Wine .....	25
RS-9: Edit Shelf .....	26
IRS (Inventory and Recommender Systems Crossover) Use Cases .....	27-28
IRS-1: From Recommender to Inventory.....	27
IRS-2: Autocomplete.....	28
FP (Front Page) Use Cases .....	29-34
FP-1: Popular Wines .....	29
FP-2: Login Account.....	30
FP-3: Register an Account.....	31
FP-4: Edit an Account.....	32
FP-5: Delete an Account .....	33
FP-6: Logout.....	34

IWP (Individual Wine Pages) Use Cases .....	<b>35-37</b>
IWP-1: Detailed Wine Information .....	35
IWP-2: Report System Abuse .....	36
IWP-3: View Average Price .....	37

## Use Case Information Tables

	Use Case Groupings
<b>IS</b>	Use case for Inventory System
<b>RS</b>	Use case for Recommender System
<b>IRS</b>	Use case for both Inventory and Recommender Systems
<b>FP</b>	Use case is for the Front Pages, seen by the user before they choose to access their inventory or recommendations
<b>IWP</b>	Use case for Specific Wine Pages that can be accessed from both the inventory system and the recommender system

	Priority
<b>1</b>	Highest Priority – Vital to Application
<b>2</b>	Medium Priority – Desirable, but not essential
<b>3</b>	Low Priority – Implement time permitting

	Progress Status
<b>Planning</b>	Use case is being considered for project.
<b>Designing</b>	Details of use case are being determined.
<b>Developing</b>	Use case is being programmed.
<b>Testing</b>	Use case is being tested to ensure it is working as designed.
<b>Completed</b>	Use case is done. There is no more work to do.
<b>Deferred</b>	Development of use case is for future release.

<b>IS-1: Add a Wine</b>	
<b>Description:</b>	This Use Case outlines the ability for a User to add a Bottle of Wine to their Inventory.
<b>Desired Outcome:</b>	The outcome of this Use Case is to allow the User to add a Bottle of Wine to their Inventory.
<b>User Goals:</b>	<p>The User wants to be able to add a specific Bottle of Wine to their Inventory, wherein:</p> <ol style="list-style-type: none"> <li>1. Entering information about the Bottle of Wine (i.e. Wine Name, Wine Type, Varietal, Vintage, Place of Origin, Price, Quantity).</li> <li>2. Entering information about the User's interpretation of the Bottle of Wine (i.e. Rating, Recommendation, Notes, Tasting Attributes).</li> <li>3. Choosing a Storage Location for that Bottle of Wine,</li> <li>4. Auto-completing the information from matching wines already in the database.</li> </ol>
<b>Primary Actor:</b>	User of Application
<b>Dependency Use Cases:</b>	FP-2: Login Account
<b>Requirements:</b>	<p>S/R-1: Store Wine Bottle  S/R-2: Store Wine Attributes  S/R-4: Store User Ratings  S/R-5: Store User Reviews/Descriptions</p>
<b>Details:</b>	<p>Priority Level: 1  Status: Completed</p>
<b>Pre-conditions:</b>	<ol style="list-style-type: none"> <li>1. The User shall have created an Account.</li> <li>2. The User shall be authenticated via that Account with the Application.</li> <li>3. The Account shall have at least one Storage Location, and at least one Storage Unit within that Storage Location.</li> </ol>
<b>Post-conditions:</b>	<ol style="list-style-type: none"> <li>1. The User's Inventory shall reflect the addition of the Bottle of Wine.</li> </ol>
<b>Trigger:</b>	<ol style="list-style-type: none"> <li>1. The User shall select to add a bottle of wine.</li> </ol>
<b>Workflow:</b>	<ol style="list-style-type: none"> <li>1. The user shall log in to the application.</li> <li>2. The user shall be directed to the inventory.</li> <li>3. The user shall request to add a wine to the inventory.</li> <li>4. The user shall be presented with a form to fill out wine info.</li> <li>5. The user shall enter wine info into the form and request to save the new wine.</li> </ol>
<b>Alternate Paths:</b>	None

## IS-2: Add a Storage Location

Description:	This Use Case outlines the ability for a User to add a Storage Location to their Inventory.
Desired Outcome:	The outcome of this Use Case is to allow the User to add a Storage Location to their Inventory.
User Goals:	The user can add a storage location and associated location image to their inventory.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account
Requirements:	S/R-3: Store Storage Location S/R-7: Store Storage Location Attributes
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The User shall have created an account.</li> <li>2. The User shall be logged into their account.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The User's Inventory shall reflect the addition of the Storage Location in a sidebar.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The User shall click a button to add a storage location.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall log in to the application.</li> <li>2. The user shall be directed to the Inventory page.</li> <li>3. The user shall request to add a storage location.</li> <li>4. The user shall type into the form a location name and select a location image from their computer.</li> </ol>
Alternate Paths:	None

<b>IS-3: Edit a Storage Location</b>	
<b>Description:</b>	This use case outlines the ability for a user to edit a storage location in their inventory.
<b>Desired Outcome:</b>	The outcome of this use case is to allow the user to edit a storage location in their inventory.
<b>User Goals:</b>	The user wants to be able to edit a specific storage location in their Inventory where they can modify the name of the storage location.
<b>Primary Actor:</b>	User of Application
<b>Dependency Use Cases:</b>	IS-2: Add a Storage Location FP-2: Login Account
<b>Requirements:</b>	Misc-1: Modify Storage Location
<b>Details:</b>	Priority Level: 1 Status: Completed
<b>Pre-conditions:</b>	<ol style="list-style-type: none"> <li>1. The user shall have created an account.</li> <li>2. The user shall be logged into their account.</li> <li>3. The user will have at least one location in their inventory.</li> </ol>
<b>Post-conditions:</b>	<ol style="list-style-type: none"> <li>1. The user will see their newly added location in their inventory.</li> </ol>
<b>Trigger:</b>	<ol style="list-style-type: none"> <li>1. The user shall select to edit a storage location.</li> </ol>
<b>Workflow:</b>	<ol style="list-style-type: none"> <li>1. The user shall log in to the application.</li> <li>2. The user shall be given their inventories page.</li> <li>3. The user shall select a storage location.</li> <li>4. The user shall edit a storage location.</li> </ol>
<b>Alternate Paths:</b>	None

## IS-4: Editing Entries

Description:	This use case outlines the user's ability to edit info about their wines in their inventory.
Desired Outcome:	The user will be able to change information about their wines and that change will be reflected on the webpage.
User Goals:	The user shall desire to edit parts of a wine entry in their inventory.
Primary Actor:	User of Application
Dependency Use Cases:	IS-1: Add a Wine FP-2: Login Account
Requirements:	Misc-2: Modify Wine Attributes
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be logged in.</li> <li>2. The user shall have wine entries to edit.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The system shall store the updated information about the edited wine.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The user shall select an edit button in their inventory.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall select a wine.</li> <li>2. The user shall select to edit their chosen wine.</li> <li>3. The user shall see a form with the current wine information already filled in.</li> <li>4. The user shall change the fields that should have different information.</li> <li>5. The user shall submit the form.</li> </ol>
Alternate Paths:	None

<b>IS-5: Importing/Exporting Inventory</b>	
<b>Description:</b>	This use case outlines the user's ability to import/export an inventory from an external source like XML or CSV.
<b>Desired Outcome:</b>	The user can obtain a copy of their inventory that can be accessed offline, or can upload their inventory
<b>User Goals:</b>	To obtain an offline copy of their inventory or upload a copy of their inventory to the service
<b>Primary Actor:</b>	User of Application
<b>Dependency Use Cases:</b>	FP-2: Login Account
<b>Requirements:</b>	BS-1: Create XML File BS-2: Export XML File BS-3: Upload XML File BS-4: Make Inventory from XML File
<b>Details:</b>	Priority Level: 3 Status: Deferred
<b>Pre-conditions:</b>	<ol style="list-style-type: none"> <li>1. User shall be logged in.</li> <li>2. User shall have entries in his/her inventory OR</li> <li>3. User shall have a copy of his/her inventory to be uploaded.</li> </ol>
<b>Post-conditions:</b>	<ol style="list-style-type: none"> <li>1. The user shall be able to view an offline copy of the inventory (as an XML, CSV, or text file) OR</li> <li>2. The user shall be able to view the uploaded copy integrated into the user's inventory in the service.</li> </ol>
<b>Trigger:</b>	<ol style="list-style-type: none"> <li>1. The user shall select a "Download inventory" button OR the user shall select an "Upload inventory" button.</li> </ol>
<b>Workflow:</b>	<ol style="list-style-type: none"> <li>1. The user shall select "Download inventory" button (or "Upload inventory")</li> <li>2. The system shall display option for download as XML, CSV, or other type (or interface for uploading inventory)</li> <li>3. The user shall select file format (or upload file)</li> <li>4. The system shall provide file in specified format (or upload chosen file to database)</li> </ol>
<b>Alternate Paths:</b>	None



## IS-6: Removing Entries

Description:	This use case outlines the user's ability to remove some or all bottles of a particular wine in the inventory.
Desired Outcome:	Entries specified by the user are removed from the user's inventory.
User Goals:	To remove wines they no longer have in their inventory.
Primary Actor:	User of Application
Dependency Use Cases:	IS-1: Add a Wine FP-2: Login Account
Requirements:	S/R-9:Delete Wines
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be logged in.</li> <li>2. The user shall have entries in their inventory.</li> <li>3. The user shall have entries they wish to remove from the inventory.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The user shall see that the specified entries have been removed from the inventory.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The user shall click on a "delete entry" button</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall select a wine to remove from their inventory.</li> <li>2. The user shall select the "delete wine" button.</li> <li>3. The user shall confirm that they want to remove the wine from their inventory.</li> <li>4. The user shall see the wine is no longer in their inventory.</li> </ol>
Alternate Paths:	None

## IS-7: Viewing Statistics

Description:	This use case outlines the user's ability to view detailed statistics about their inventory or specific wines in their inventory.
Desired Outcome:	The user shall be able to see how many wines they have in their inventory, how many of those wines are red, how many of those wines are white, and how many locations they have.
User Goals:	To view inventory statistics.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account
Requirements:	Di-4: Display Wine Statistics Di-5: Display Inventory Statistics
Details:	Priority Level: 1 Status: Deferred
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall be presented with statistics on their profile modal page.
Trigger:	1. The user shall select to view their profile.
Workflow:	1. The user shall be logged in. 2. The user shall select the top navigation bar button to view their profile. 3. The user shall see their inventory statistics on their profile modal. 4. The user shall select the "Close" button when finished.
Alternate Paths:	None

## IS-8: Filter Inventory

Description:	This use case outlines the ability for the user to filter their wine inventory by selecting from various attributes of wine and by user generated tags.
Desired Outcome:	The user shall be able to filter their wines in their inventory.
User Goals:	The user's wines are filtered so they are no longer viewing their whole inventory.
Primary Actor:	User of Application
Dependency Use Cases:	IS-1: Add a Wine FP-2: Login Account
Requirements:	S/S-1: Filter Wines S/S-2: Sort Wines by Attribute
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in. 2. The user shall have an inventory containing one or more wines.
Post-conditions:	1. The user shall see their filtered inventory.
Trigger:	1. The user selects a filter button.
Workflow:	1. The user shall be viewing the inventory page. 2. The user shall select one or more filters. 3. The user shall be presented with their inventory with that filter applied.
Alternate Paths:	None

## IS-9: Archive/History of Inventory

Description:	This use case outlines the ability for the user to view their archive/history of their inventory, so they can see the transactions that have been made in their inventory, like added and removed wines.
Desired Outcome:	The user shall be able to view their archive/history of added or removed wines from their inventory.
User Goals:	The user wants to view their archive/history of their wine inventory.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account IS-1: Add a Wine
Requirements:	S/S-3: Wine History – Everything S/S-4: Wine History – Recently Added S/S-5: Wine History – Removed
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	1. The user shall be logged in. 2. The user shall have had previous interactions with the inventory.
Post-conditions:	1. The system shall move a page with the archive/history to the user.
Trigger:	1. The user shall click on the inventory button to view inventory. 2. The user shall click on the view archive/history button.
Workflow:	1. The user shall be signed in. 2. The user shall be viewing their inventory. 3. The user shall request the wine history feature.
Alternate Paths:	None

## IS-10: Search Inventory

Description:	This use case outlines the ability for the user to search their inventory for specific wines by typing or by voice.
Desired Outcome:	The user shall be able to search for wines by name.
User Goals:	The user wants to find the wines they are looking for quickly.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account IS-1: Add a Wine
Requirements:	S/S-6: Search Database for Wine – Name S/S-7: Search Database for Wine – Attribute S/S-8: Search Inventory for Wine – Name S/S-9: Search Inventory for Wine – Name
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall be presented with wines matching the user's search terms.
Trigger:	1. The user shall search in a field or the user shall click the microphone button to begin speaking a voice search.
Workflow:	1. The user shall be on their inventory page. 2. The user shall click inside the search box. 3. The user shall input search terms for a wine. 4. The user shall be presented with wines matching their search terms.
Alternate Paths:	1. The user shall be on their inventory page. 2. The user shall click on the microphone button. 3. The user shall speak wine search terms. 4. The system shall populate the text search field with the spoke search terms. 5. The user shall be presented with wines matching their spoken search terms.

## IS-11: Wine Wishlist

Description:	Outlines the ability for the user to add wines to a wishlist of wines they would like to have.
Desired Outcome:	User shall be able to access this list quickly and easily modify this list as desired
User Goals:	To keep track of wines that they would like to have in the future, but currently are not able to acquire
Primary Actor:	User of Application
Dependency Use Cases:	None
Requirements:	S/R-10: Add Item to Wishlist S/R-11: Delete Item from Wishlist Di-11: Display Wishlist
Details:	Priority Level: 2 Status: Completed
Pre-conditions:	1. The user shall be logged in to their account
Post-conditions:	1. The user shall be able to view wines they wish to own.
Trigger:	1. The user shall select the wishlist filter in their inventory.
Workflow:	1. The user shall navigate to their inventory. 2. The user shall select the wishlist filter. 3. The user shall be presented with the wines on their wishlist.
Alternate Paths:	None

## IS-12: Inventory Security

Description:	This use case outlines the ability for the user to choose the level of privacy of showing their wine inventory by having access to a security setting (i.e. self, friends, public). Default will be at full privacy.
Desired Outcome:	The user shall be able to edit their security setting to select the level of privacy of their wine inventory.
User Goals:	The user wants to have a security setting to limit who can view their inventory.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account
Requirements:	PP-4: Privacy Settings – Update Di-6: Display Privacy Settings
Details:	Priority Level: 3 Status: Deferred
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall have their inventory privacy setting set.
Trigger:	1. The user shall click on the inventory button to view inventory. 2. The user shall click on the edit security button.
Workflow:	1. The user shall be logged in and be viewing their inventory. 2. The user shall click on the edit security button. 3. The user shall be presented with the list of privacy levels. 4. The user shall select one of the privacy options. 5. The user shall confirm to update their security settings. 6. The user shall be notified their security settings have been updated.
Alternate Paths:	5. The user shall decide to not to edit their security settings by pressing the cancel button. 6. The user shall be returned to the inventory page.

## IS-13: Rate and Review

Description:	The user shall be able to rate and review a wine in their inventory.
Desired Outcome:	The user shall have submitted a rating and review of the wine which is stored in their inventory.
User Goals:	The user wants to rate and review a wine.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account IS-1: Add a wine OR IS-4: Editing Entries
Requirements:	Di-2: Display User Ratings Di-3: Display User Reviews
Details:	Priority Level: 1 Status: Deferred
Pre-conditions:	1. The user is logged into the application 2. The user is viewing the inventory page.
Post-conditions:	1. The user's rating and/or review shall be stored in their inventory entry for that wine.
Trigger:	1. The user shall request to add or edit a wine to their inventory.
Workflow:	1. The user shall be logged in and be viewing the inventory page. 2. The user shall request to add a wine to their inventory. 3. The user shall be presented with a form to enter information about the wine. 4. The user shall enter rating and/or review information in the form, as well as the required wine data and any optional fields of the user's choice. 5. The user shall confirm that the wine is ready to be added to their inventory. 6. The user shall be presented with their updated inventory, which now contains the new wine with their rating and/or review.
Alternate Paths:	1. The user shall be logged in and be viewing the inventory page. 2. The user shall request to edit a wine in their inventory. 3. The user shall be presented with a form to edit the selected wine's stored information. 4. The user shall enter rating and/or review information in the edit form. 5. The user shall confirm that the wine information is ready to be updated. 6. The user shall be presented with their updated inventory, which contains the edited wine with their new rating and/or review.



## IS-14: Remove an Inventory Location

Description:	The user shall be able to remove a location from their inventory.
Desired Outcome:	The user shall have removed an inventory location.
User Goals:	The user shall desire to remove an inventory location.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account IS-2: Add a Storage Location
Requirements:	S/R-12: Delete an Inventory Location
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged into the application 2. The user shall be viewing the inventory page.
Post-conditions:	1. The user shall see the location is removed.
Trigger:	1. The user shall request the remove location feature.
Workflow:	1. The user shall be logged in and be viewing the inventory page. 2. The user shall request to remove a location. 3. The user shall confirm that they want to remove the location. 4. The user shall see the location is removed.
Alternate Paths:	None

<b>RS-1: View Recommendations</b>	
Description:	Outlines how a user may view their recommendations from the recommender system.
Desired Outcome:	User shall be able to view recommended wines and get more recommendations.
User Goals:	To be recommended wines that have a high likelihood of the user enjoying.
Primary Actor:	The System
Dependency Use Cases:	None
Requirements:	HP-5: View Random Wine HP-6: View Random Wine Info RS-11: Shuffle Shelf RS-14: Generate Recommendations
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in
Post-conditions:	1. The user shall see recommendations of wines.
Trigger:	1. The user shall navigate to their recommender page.
Workflow:	1. The user shall click on a button to take them to their recommender page. 2. The user shall see recommended wines from their default shelf. 3. The user shall select a button to view more recommendations from that shelf.
Alternate Paths:	1. The user shall click on a button to take them to their recommender page. 2. The user shall select one of their non-default shelves. 3. The user shall see recommended wines from their non-default shelf. 4. The user shall select a button to view more recommendations from that shelf.

## RS-2: Shuffle Channels

Description:	The user may choose to receive recommendations from one or more channels simultaneously. A channel is chosen at random and its recommendations are displayed.
Desired Outcome:	The user is shown a random wine recommendation.
User Goals:	The user may desire to view a wine recommendation from any of their recommendation shelves.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account RS-1: View Recommendations
Requirements:	RS-11: Shuffle Shelf
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be logged in.</li> <li>2. The user shall be viewing their recommendation shelves.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be presented with recommendations from a random recommendation shelf.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The user shall select the shuffle option on the recommender page.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be logged into the application and viewing the recommender page.</li> <li>2. The user shall request the shuffle feature.</li> <li>3. The user shall be presented with recommendations from a recommendation shelf chosen at random.</li> </ol>
Alternate Paths:	None

<b>RS-3: Buy Wine</b>	
Description:	The user shall be able to directly purchase recommended wines.
Desired Outcome:	The user shall be redirected to a site which will facilitate the wine purchase.
User Goals:	The user may desire to purchase a recommended wine without the burden of locating the same wine on a different site.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account RS-1: View Recommendations
Requirements:	Di-7: Display External Link
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	1. The user shall be logged in. 2. The user shall have been recommended a wine.
Post-conditions:	1. The user shall be redirected to a page where they may purchase the wine.
Trigger:	1. The user shall select the “Buy Now” feature.
Workflow:	1. The user shall be logged in and is viewing recommendations. 2. The user shall request the “Buy Now” feature. 3. The user shall be redirected to a page where they may purchase the wine.
Alternate Paths:	None

<b>RS-4: Buddy Recommendations</b>	
Description:	The user would like to be able to view their friend's recommender shelves.
Desired Outcome:	The user may view their friend's recommendation shelves.
User Goals:	The user may desire to view their friend's recommendation shelves.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account RS-1: View Recommendations
Requirements:	HP-2: Login to System RS-1: Create Shelf RS-2: Export Shelf Seed RS-3: Create Shelf – Preexisting Seed RS-9: View Current Shelf
Details:	Priority Level: 3 Status: Deferred
Pre-conditions:	1. The user shall be logged in 2. The user shall have friend(s) with accounts.
Post-conditions:	1. The user shall be able to view their friend's recommendation shelf.
Trigger:	1. The user shall request the "View Friend's Recommendations" feature.
Workflow:	1. The user shall be logged in and is viewing a friend's profile. 2. The user shall request to view the friend's recommendation shelves. 3. The user shall be presented with a list of shelves the friend has given them permission to view. 4. The user shall select a recommendation shelf to view. 5. The user shall receive recommendations from their friend's shelf, and may also view the channel's seed wines.
Alternate Paths:	None

<b>RS-5: Create a Shelf</b>	
Description:	The user shall be able to create a recommendation shelf seeded with one or more wines.
Desired Outcome:	A recommendation shelf is created based on the wine(s) the selected seed wines, and the new shelf is added to the user's list of shelves.
User Goals:	The user shall desire to create a recommendation channel seeded with one or more wines.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account RS-1: View Recommendations
Requirements:	HP-2: Login to System RS-1: Create Shelf RS-3: Create Shelf – Preexisting Seed RS-9: View Current Shelf
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in. 2. The user shall be viewing the recommender page.
Post-conditions:	1. The user shall see that the shelf has been created and added to the user's list of recommendation shelves.
Trigger:	1. The user shall request the "Add Shelf" feature on the recommender page.
Workflow:	1. The user shall be logged in and viewing the recommender page. 2. The user shall request the "Add Shelf" feature on the recommender page. 3. The user shall specify a shelf name and select one or more seed wines. 4. The user shall confirm the create shelf request. 5. The system shall add the new shelf to the user's list of shelves. 6. The user shall be presented with the updated recommender page with access to the newly created shelf.
Alternate Paths:	None

RS-6: Remove a Shelf	
Description:	The user shall be able to remove a recommendation shelf from their recommender page.
Desired Outcome:	The selected channel shall be removed from the user's list of recommendation shelves.
User Goals:	The user shall desire to remove a recommendation shelf from their recommender page.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Account Login RS-1: View Recommendations
Requirements:	RS-4: Remove Shelf – Database RS-5: Remove Shelf – Display RS-6: Remove Shelf – Button RS-7: Store Shelves RS-8: View Shelf List (Sidebar) RS-9: View Current Shelf
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in to their account. 2. The user shall have at least one recommendation shelf.
Post-conditions:	1. The user shall see that the selected channel is absent from the user's recommendation shelves list.
Trigger:	1. The user shall select the "Delete Shelf" option while editing a recommendation shelf.
Workflow:	1. The user shall be logged in and viewing their recommendation shelves. 2. The user shall request to edit the current shelf. 3. The user shall request to delete the shelf from the edit shelf form. 4. The user shall be presented with the updated list of recommendation shelves, with the deleted shelf absent.
Alternate Paths:	None

<b>RS-7: Navigate Shelves</b>	
Description:	The user shall be able to navigate between their existing recommendation shelves.
Desired Outcome:	The user shall be presented with recommendations from the recommendation shelf of their choice.
User Goals:	The user shall desire to view recommendations from the selected recommendation shelf.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account RS-1: View Recommendations
Requirements:	RS-7: Store Shelves RS-8: View Shelf List (Sidebar) RS-9: View Current Shelf RS-10: Select Different Shelf
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged into their account 2. The user shall have more than one recommendation shelf.
Post-conditions:	1. The user shall be presented with recommendations from the selected recommendation shelf.
Trigger:	1. The user shall select a different recommendation shelf from the one they are currently viewing.
Workflow:	1. The user shall be logged in and be viewing the recommender page. 2. The user shall request recommendations from a different recommendation shelf. 3. The user shall be presented with recommendations from the selected recommendation shelf.
Alternate Paths:	None



<b>RS-8: Recommend Random Wine</b>	
Description:	The user can select to be shown a random wine, regardless of their ratings, recommender history or preferences.
Desired Outcome:	The user is shown a random bottle of wine as a recommendation
User Goals:	The user wants to be shown wine at random
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account
Requirements:	HP-5: View Random Wine HP-6: View Random Wine Info
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	1. The user shall be logged in. 2. The user shall be viewing the recommender page.
Post-conditions:	1. The user shall be presented with a random wine recommendation.
Trigger:	1. The user shall request the random recommendation feature.
Workflow:	1. The user shall be logged in and viewing the recommender page. 2. The user shall request a random wine recommendation. 3. The user shall be presented with a random wine recommendation.
Alternate Paths:	None

RS-9: Edit Shelf	
Description:	The user shall be able to change the seed wines of a shelf.
Desired Outcome:	The user shall be able to change the recommendations they receive from an existing shelf.
User Goals:	The user shall desire to change the behavior of an existing shelf.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account RS-5: Create a Shelf
Requirements:	RS-15: Edit a Shelf
Details:	Priority Level: 2 Status: Completed
Pre-conditions:	1. The user shall be logged in 2. The user shall be viewing the recommender page.
Post-conditions:	1. The user shall see different kinds of recommendations on a shelf.
Trigger:	1. The user shall request the edit shelf feature.
Workflow:	1. The user shall be logged in. 2. The user shall be viewing the recommender page. 3. The user shall request to edit a shelf 4. The user shall select different seed wines. 5. The user shall submit the altered shelf. 6. The user shall view adjusted recommendations.
Alternate Paths:	None

IRS-1: From Recommender to Inventory	
Description:	Allows the user to place wines which were recommended to them directly into their user inventory locations.
Desired Outcome:	The recommended wine is added to the user's inventory location.
User Goals:	The user may want to add a wine suggested by one of their recommendation shelves into their inventory.
Primary Actor:	User of Application
Dependency Use Cases:	IS-1: Add a Wine FP-2: Account Login RS-1: Recommendations
Requirements:	RS-12: Recommender to Inventory RS-13: Transfer Confirmation
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall have an account.</li> <li>2. The user shall have an inventory.</li> <li>3. The user shall have a recommender shelf.</li> <li>4. The user shall have been recommended a wine.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The user shall see, upon checking their inventory, that the recommended wine has been placed into the user's inventory.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The user shall request to add the recommender wine to the inventory.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be logged in and be viewing recommendations from a recommender shelf.</li> <li>2. The user shall request to add a recommended wine to their inventory.</li> <li>3. The user shall select a location to add the wine to.</li> <li>4. The user shall confirm that the wine should be added to the selected location.</li> <li>5. The system shall add the wine to the user's inventory.</li> <li>6. The user shall remain viewing the recommender page.</li> </ol>
Alternate Paths:	None

## IRS-2: Autocomplete

Description:	This use case outlines the ability for the user to select wine names based off of common wines in the inventory or recommender.
Desired Outcome:	The user shall be given a list of wines when adding a wine to their inventory or creating a new shelf in the recommender system.
User Goals:	The user is able to more quickly add a wine to their inventory or select seed wines for a recommender shelf.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account
Requirements:	S/S-6: Search Database for Wine – Name S/S-7: Search Database for Wine – Attribute S/S-8: Search Inventory for Wine – Name S/S-9: Search Inventory for Wine – Attribute Di-9: Autocomplete – Database Search Di-10: Autocomplete – Inventory Add
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall see a list of wines similar to what they have typed.
Trigger:	1. The user shall pause when typing in a text field.
Workflow:	1. The user shall be logged in and viewing the inventory page. 2. The user shall request to add a wine. 3. The user shall pause when typing in info about a wine. 4. The user shall be given a list of wines identical or similar to what they have typed.
Alternate Paths:	1. The user shall be logged in and viewing the recommender page. 2. The user shall request to add a shelf. 3. The user shall start typing in a seed wine for the shelf and pause. 4. The user shall be given a list of wines identical or similar to what they have typed.

<b>FP-1: Popular Wines</b>	
Description:	The user shall be able to view popular wines on the site's front page.
Desired Outcome:	The user shall be able to view what wines are popular among other users and the public.
User Goals:	To see what wines are widely possessed or highly rated by other users.
Primary Actor:	User of Application
Dependency Use Cases:	None
Requirements:	HP-1: Homepage Display PP-2: Popularity Count PP-3: Popularity Sort
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	1. The user shall not be logged in. 2. The user shall be viewing the front splash page.
Post-conditions:	1. The user shall be presented with the system's popular wines on the front page.
Trigger:	1. The user shall visit the front splash page.
Workflow:	1. The user shall visit the front splash page. 2. The user shall view the "Popular Wines" section of the page.
Alternate Paths:	None

<b>FP-2: Login Account</b>	
Description:	The user shall be able to log into a previously created user account in the system using an email and password.
Desired Outcome:	The user shall be able to log into an account in the system.
User Goals:	The user will desire to log into an account in order to access the majority of the site's pages and features.
Primary Actor:	User of Application
Dependency Use Cases:	None
Requirements:	HP-2: Login to System HP-4: Login with Facebook PP-1: View Profile S/R-6: Store User Account Info
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall have a currently registered account.</li> <li>2. The user shall be viewing the front splash page.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be logged into their account and may access the rest of the system.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The user shall request to log in from the front splash page.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall not be logged in and shall be viewing the front splash page.</li> <li>2. The user shall request to log in.</li> <li>3. The user shall be presented with a form to provide an email address and password.</li> <li>4. The system shall check whether the user's input matches an existing user account in the system.</li> <li>5. Possible results: <ol style="list-style-type: none"> <li>a. If a correct email address and password pair has been entered, the user shall be logged in and redirected to their inventory page.</li> <li>b. If an incorrect email address and password pair has been entered, the user shall be presented with an error message and may reenter their information.</li> </ol> </li> </ol>
Alternate Paths:	None

## FP-3: Register an Account

Description:	The user shall be able to create a new user account in the system, using an email and password for logging in to the new account.
Desired Outcome:	The user shall be able to create a new account in the system.
User Goals:	The user shall desire to be able to create an account.
Primary Actor:	User of Application
Dependency Use Cases:	None
Requirements:	HP-3: Create Account
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	3. The user shall not be logged in. 4. The user shall be viewing the front splash page, which has a login option.
Post-conditions:	1. The user shall be logged into their new account and can access the rest of the system.
Trigger:	1. The user shall request to register an account from the front splash page.
Workflow:	6. The user shall not be logged in and viewing the front splash page. 7. The user shall request to register an account. 8. The user shall be presented with a form to provide an email address, password, display name, location, birthday, and profile image. 9. The system shall check whether the user's inputs are valid. 10. The user shall be redirected to their inventory with a new account created if there are no invalid credentials and the email has not been taken.
Alternate Paths:	1. The user shall not be logged in and viewing the front splash page. 2. The user shall request to register an account. 3. The user shall be presented with a form to provide an email address, password, display name, location, birthday, and profile image. 4. The system shall check whether the user's inputs are valid. 5. The user shall be notified with the corresponding error message if the user puts in incorrect credentials.

## FP-4: Edit an Account

Description:	The user shall be able to modify their account information, which can be their password, display name, location, birthday, and/or profile image.
Desired Outcome:	The user shall be able to have their account information modified to the user's preferences.
User Goals:	The user wants to be able to edit their user account.
Primary Actor:	User of Application
Dependency Use Cases:	FP-3: Create an Account
Requirements:	PP-5: Edit User Account
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall have their account modified and is redirected to the inventory page.
Trigger:	1. The user shall request to edit an account from the profile window.
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be logged in and viewing the profile window.</li> <li>2. The user shall request to edit an account.</li> <li>3. The user shall be presented with a form to provide their password, and optionally their new password, display name, location, birthday, and/or profile image.</li> <li>4. The system shall check whether the user's inputs are valid.</li> <li>5. The user shall be redirected to their inventory page and their account updated if there are no invalid credentials.</li> </ol>
Alternate Paths:	<ol style="list-style-type: none"> <li>1. The user shall be logged in and is viewing the profile window.</li> <li>2. The user shall request to edit an account.</li> <li>3. The user shall be presented with a form to provide their password, and optionally their new password, display name, location, birthday, and/or profile image.</li> <li>4. The system shall check whether the user's inputs are valid.</li> <li>5. The user shall put in incorrect credentials and is notified with the corresponding error message.</li> </ol>



<b>FP-5: Delete an Account</b>	
Description:	The user shall be able to delete their account from the system.
Desired Outcome:	The user shall be able to have their account deleted.
User Goals:	The user shall desire to be able to delete their user account.
Primary Actor:	User of Application
Dependency Use Cases:	FP-3: Create an Account
Requirements:	PP-6: Delete User Account
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall have their account deleted and is redirected to the front splash page.
Trigger:	1. The user shall request to delete their account from the profile window.
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be logged in and is viewing the profile window.</li> <li>2. The user shall request to delete their account.</li> <li>3. The user shall be presented with a message asking for confirmation.</li> <li>4. The user shall confirm their request.</li> <li>5. The system shall delete the user's account.</li> <li>6. The user shall be redirected to the front splash page.</li> </ol>
Alternate Paths:	None

<b>FP-6: Logout</b>	
Description:	The user shall be able to log out from their account.
Desired Outcome:	The user shall be able to log out and be sent to the front splash page.
User Goals:	The user shall desire to be able to log out.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account FP-3: Create an Account
Requirements:	Misc-6: Logout User
Details:	Priority Level: 1 Status: Completed
Pre-conditions:	1. The user shall be logged in.
Post-conditions:	1. The user shall be logged out from the system and is redirected to the front splash page.
Trigger:	1. The user shall request to log out from their account from either the inventory or recommender page.
Workflow:	1. The user shall be logged in and is either on the inventory or recommender page. 2. The user shall request to log out from their account. 3. The system shall log out the user's account. 4. The user shall be redirected to the front splash page.
Alternate Paths:	None

<b>IWP-1: Detailed Wine Information</b>	
Description:	The user is given information about a wine on that wine's individual page
Desired Outcome:	The user is presented with additional information about the wine they are viewing.
User Goals:	The user will obtain the following information about the wine: wine name, winery, varietal, region, vintage, user description/review, and image.
Primary Actor:	User of the application
Dependency Use Cases:	FP-2: Login Account
Requirements:	Di-1: Display Wine Attributes
Details:	Priority Level: 2 Status: Completed
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be logged in</li> <li>2. The user shall be on a page featuring one or more interactive wine representations</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be presented with all relevant information associated with that wine in the system.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>1. The user shall click on an interactive wine representation, such as an image or list item.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be logged in and be viewing a page with one or more wine listings.</li> <li>2. The user shall request additional information about the wine by interacting with its representation on the current page.</li> <li>3. The user shall be presented with the information about the selected wine.</li> </ol>
Alternate Paths:	None

## IWP-2: Report System Abuse

Description:	This use case outlines the ability for the user to report system abuse of other users (inappropriate names/pictures)
Desired Outcome:	The user shall be able to flag existing wines for moderation by an administrator.
User Goals:	The user wants to maintain a mature and high quality wine database for his or her personal use as well as for others.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account IWP-1: Detailed Wine Information
Requirements:	Misc-4: Recognize System Abuse Misc-5: Remove System Abuse
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	<ol style="list-style-type: none"> <li>1. The user shall be logged in.</li> <li>2. The user shall be viewing additional information about a wine.</li> </ol>
Post-conditions:	<ol style="list-style-type: none"> <li>1. The system shall send a report to the administrator notifying them of a possible system abuse.</li> </ol>
Trigger:	<ol style="list-style-type: none"> <li>2. The user shall request to report a wine from its detailed information presentation.</li> </ol>
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be viewing additional information about a wine.</li> <li>2. The user shall request to report the current wine.</li> <li>3. The system shall send a report to the administrator requesting that the wine be reviewed for system abuse.</li> </ol>
Alternate Paths:	None

<b>IWP-3: View Average Price</b>	
Description:	This use case outlines the ability for a user to view the average price of a wine.
Desired Outcome:	The user shall be presented with the average price of the wine.
User Goals:	The user wants to view the average price of a particular wine.
Primary Actor:	User of Application
Dependency Use Cases:	FP-2: Login Account IWP-1: Detailed Wine Information
Requirements:	Di-8: Display Average Price Misc-3: Calculate Average Price
Details:	Priority Level: 2 Status: Deferred
Pre-conditions:	1. The user shall be logged into their account
Post-conditions:	1. The system shall display the average price of a particular
Trigger:	1. The user shall request additional information about a wine.
Workflow:	<ol style="list-style-type: none"> <li>1. The user shall be logged in and be viewing a page with one or more wine listings.</li> <li>2. The user shall request additional information about the wine by interacting with its representation on the current page.</li> <li>3. The user shall be presented with the average price of the wine, along with other all other information presented in a detailed wine information page.</li> </ol>
Alternate Paths:	None