 Requirements

**Paywall**

Robert J. Muller

December 6, 2013

# Introduction

The business of Phoenix Bioinformatics is to provide scientific data in a sustainable manner. To do so, we have determined that we need to charge scientists for access to the products we offer. To better support this approach, we want to develop a basic paywall system.

A *paywall* is a system that prevents Internet users from accessing webpage content without a paid subscription (<http://en.wikipedia.org/wiki/Paywall>). A *hard paywall* is a paywall that prevents any access without subscription; a *soft paywall* is more flexible, permitting certain kinds of unrestricted or partially restricted access without a paid subscription. Phoenix will support six kinds of access:

* Free content: web pages accessible by anyone with no charge
* Paid content: web pages requiring a subscription and authentication
* Metered content: web pages allowing a certain number of accesses per month before requiring a subscription (determined by IP address)
* Community content: web pages accessible only to a logged-in user with a community id
* Curator content: web pages accessible only to authenticated curators
* Administrator content: web pages accessible only to authenticated administrators

Each of these types of access has a use case in this document as the entry point into the system.

A full-scale, commercial-quality paywall offers lots of features, including subscription management, access control, system monitoring, and system and financial reporting. It is also very expensive, usually based on a percentage of subscription revenue between 10 and 20%.

Because of some of the decisions we've made about our subscription model, we do not really need a comprehensive paywall solution. This document defines the specific use cases we need to support to achieve the level of access control to our content we want.

This document also includes the use cases for basic registration and for subscription processing, but not for extended lab and personal registration and profile management or for invoice processing or other financial processes, which are outside the scope of the paywall. Also note that the assigning of Curator and Administrator roles to individual users is done outside the scope of this system.

# Allow Free Access

Certain TAIR content is available to any Internet user without login.

1. User submits a www.arabidopsis.org URL that corresponds to one of the following:
   1. A Servlet class designed to provide free content
   2. A URL that redirects to a free content URL or system
   3. A URL specifically designated as free content in the access database
2. System responds by showing the content.

# Allow Paid Access

Certain TAIR content is available to any registered user who has paid for a subscription, either through individual or group subscription or through institutional or company subscription.

1. User submits a www.arabidopsis.org URL that corresponds to one of the following:
   1. A Servlet class designed to provide paid content
   2. A URL that redirects to a paid-content URL or system
   3. A URL specifically designated as paid content in the access database
2. System uses the Check Subscription use case to validate the subscription and shows the content.

Extensions:

* If the Check Subscription use case fails, the System treats the user as an unpaid user and uses the Subscribe use case to allow the user to subscribe.

# Allow Metered Access

For some paid content, TAIR allows a User a certain number of accesses per month (a technique called metering). This permits the User to see the restricted pages without logging in or paying a subscription.

1. User submits a www.arabidopsis.org URL that corresponds to one of the following:
   1. A Servlet class designed to provide metered content
   2. A URL that redirects to a metered-content URL or system
   3. A URL specifically designated as metered content in the access database
2. System uses the Check Subscription use case to validate the subscription and fails. System gets the IP address of the User and retrieves the number of metered accesses for the user during the current month. The System also gets the meter limit for the URL (based on the subsystem that contains the content to which the URL refers). If the metered accesses is less than or equal to the limit, the System shows the content.

Extensions:

* If the number of metered accesses is two fewer than the limit, System displays an alert informing the user that the user must subscribe after 10 accesses.
* If the number of metered accesses is greater than the limit, System uses the Subscribe use case to let the User subscribe.
* At midnight on the first day of every month, System clears the metering database.

# Allow Community Access

For some content, TAIR allows only an authenticated community member to access the page. An authenticated community member has no special access to content other than community content.

1. User uses the Login use case and is granted the Community role.
2. System displays the content.

Extensions:

* If User is not an authenticated Community member, System displays an error page telling the user that the content is not accessible and lets the user try again.
* If the content is metered or paid or curator or administrator content, System uses those use cases to determine whether to display the content.

# Allow Curator Access

For some content, TAIR allows only an authenticated curator to access the page. An authenticated curator can also access any free, metered, or paid content.

1. User uses the Login use case and is granted the Curator role.
2. System displays the content.

Extensions:

* If User is not an authenticated Curator, System displays an error page telling the user that the content is not accessible.

# Allow Administrator Access

For some content, TAIR allows only an authenticated administrator to access the page. An authenticated administrator can access any content.

1. User uses the Login use case and is granted the Administrator role.
2. System displays the content.

Extensions:

* If User is not an authenticated Administrator, System displays an error page telling the user that the paid, curator-accessible, or administrator-accessible content is not accessible.

# Check Subscription

Under various circumstances, the System needs to check whether a user is a subscriber. In various ways, the user authenticates (that is, uses the Login use case) and provides an IP address identifying the system they're using to submit requests (this is part of the HTTP package for the request). The System then determines their subscription status. This is a System use case that other use cases use by reference.

1. System retrieves the user's subscription expiration date using the IP address and username. If the expiration date is greater than the current date, the System validates the subscription.

Extensions:

* If the IP address or username is not found, the System invalidates the subscription.
* If the expiration date is less than or equal to the current date, the System invalidates the subscription.

# Login

TAIR users log into the TAIR system to authenticate and to obtain various categories of access (roles).

1. The User logs in either by submitting a username and password.
2. The System checks to see whether the user is already logged in with a different IP address and logs the User in if not, setting the community id for the user and granting the user the Community role.

Extensions:

* If the username or password does not match that of a registered user, the System does not log the User in and uses the Subscribe use case.
* The User can login by logging into Facebook, Twitter, or LinkedIn through Oauth (<http://en.wikipedia.org/wiki/OAuth>). If the User uses Oauth, the System receives the authorization response and requests the User to authorize access of their information from the responding system. If the User responds affirmatively, the System logs in the User, uses the Check Subscription use case to validate the subscription, and grants the Community role.
* The User can use an IP address that is a valid organizational subscription, meaning that a company or institution has subscribed and supplied a range of IP addresses. If the User does not login with a username and password or through OAuth, the System checks the provided IP address against the database of subscribed IP ranges and validates the subscription and logs in the User and grants the Community role if the IP address is there or invalidates it if it isn't. The System does not set the community id for the User, who must login with username and password to have that done.
* If the User enters a username and password but is already logged in with a different IP address, the System does not log the User in and displays the Invalid Login Attempt error page. This prevents multiple users from sharing a username and password.
* If the System validates the subscription and the User is a member of a lab group, the user gets the lab role (identified by the lab name).
* Certain users are registered in the current TAIR database, primarily for enabling ordering of seeds through ABRC. If a user is registered as a TAIR user, the System should find them already in the authentication database. This may require loading current information from the TAIR Community, Person, Organization, and Affiliation tables into a separate authentication system. Community is the superclass for Person and Organization, all of which share the community\_id column (the JOIN column). Affiliation is a many-to-many linking table that contains the community\_id of the Person and the organization\_id of the linked Organization, usually a laboratory. This corresponds to the groups discussed in the Subscribe Group use case. Note: only communities (persons or organizations) with community is\_obsolete = 'F' and status = 'active' should be available to login checking.

# Subscribe

The new User subscribes to TAIR for a year or two years.

1. System displays the Subscribe to TAIR page with required fields for username, password, re-enter password, and email. There is a checkbox for automatic renewal, checked by default. The page specifies the subscription fee. A choice field lets the User select one or two years with a default of 2 years. System updates the fee based on the choice ($9.50/month, $95/year, $190/2 years).
2. The User fills out and submits the form. System registers the user, allocates a community id, and stores the user registration and IP address for the user. System uses the Accept Payment use case to get payment from the User.

Extension:

The User can choose to subscribe a group of users (a checkbox). On doing so, System uses the Subscribe Group use case.

# Subscribe Group

The user subscribes a group of users (for a discounted rate).

1. System presents a form with fields for the user's username, password, re-enter password, email address, group name, and a table of similar fields for the multiple users in the group. A choice field lets the User select one or two years default 2 years. All fields are required.
2. User enters the group leader information (PI information) and enters one row in the table for each additional subscription desired. As User completes each row, System updates the total amount of the subscription fee with discount based on the subscription period choice and the number of rows. User submits the form.
3. System registers the PI User with the information provided and the IP address for the User. System registers the other users without IP address. System uses the Accept Payment use case to get payment from the User.

# Accept Payment

The User pays for a subscription using a credit card. The System accomplishes this by passing through to an external payment system that handles the credit card processing.

1. User chooses to pay for a subscription. System displays the credit card entry form.
2. User enters data into the form as required and submits. System transmits the data and processes the payment, then receives confirmation.
3. System uses the Confirm Payment use case to inform the User that the payment was successful and to provide a receipt.

# Confirm Payment

The System confirms the details of a subscription payment to the User.