CMS integration with Scrape and Drape

This page is designed to be read/used by CMS vendors for clients using the MatrixMaxx AMS system.

Internal Matrix: To find out which clients have Scrape and Drape and Which have static headers, refer to the Matrix Maxx Clients list.

Introduction

The MatrixMaxx AMS system generates a variety of pages meant for use on the WWW website of the association. These pages are 'ready to go' and all the association needs to do is link to them. (There is no iframing, no API pulls needed to the CMS. These are usable website pages.) These pages include:

- Login page, and associated Reset Password/Create Profile functionality
- Manage/Edit individual profile, including a variety of 'My account' functionality that overlaps the store, lists, committees, and meetings
- Manage/Edit company profile
- Committee rosters
- Membership Directories
- Meeting-related pages, including upcoming events calendar, meeting view pages, and meeting registration forms
- Commerce-related pages, including a store and cart
- Survey pages
- Sponsorship pages
- Donation pages
- Tradeshow pages

Ideally, the user shouldn't notice the difference between a CMS-generated website page and an AMS-generated website page, so the WWW website pages generated by the MatrixMaxx AMS are designed to be styled to match the rest of the CMS/Website pages in one of two ways:

- The MatrixMaxx AMS can dynamically pull the design from the CMS. This presents a higher up-front setup cost for the AMS/CMS, as additional work must be undertaken by the CMS team, but is a worthwhile investment if the association plans to be constantly changing navigation or design.

- The MatrixMaxx team can place static versions of the design/navigation/styling into our coding during implementation. This is relatively fast and easy, but also static. **This is rarely done anymore; it is typically only used if the 3rd party system can only support static headers.** Example: Client changes "About" section to be called "About Us", and they then have to call/pay the MatrixMaxx team to make this same change on our pages.

Most associations want the dynamic wrapping/styling, and so the Scrape & Drape module in MatrixMaxx allows the AMS to reach out to the website's CMS, 'scrape' the design/styles from a special page(s) in the CMS, and then 'drape' this look onto the MatrixMaxx-generated WWW pages, thus providing a seamless user experience in terms of styling.

Technical Details

Here is what the CMS vendor/implementer and association will need to do in order to support the Scrape and Drape dynamic wrapping functionality:

- First, the CMS vendor and the Association must agree on which MatrixMaxx pages are being used/linked in the website, and the sections/stubs need to be determined based on this. **At minimum, there must be one default stub**, but many groups create multiple stubs to give the various pages different appearances.
    - For example, perhaps the association has decided to use the login page, profile pages, meeting pages, and committee pages, and wish for them to all appear on different sections of the website
    - A total of 5 stubs will therefore be necessary for this implementation (see technical details in the next section)
        - a 'Login' stub for the login page
        - a 'My Account' stub for all the profile-related pages, that puts them in this section of the website
        - an 'Event' stub for all meeting-related pages, that puts them in this section of the website
        - an 'About Us' stub for all of the committee-related pages, that puts them in this section of the website
        - a default stub, that has no designation of website section, that will be used by any other pages that the association may decide to use in the future (e.g., surveys) or if one of the main stubs is accidentally damaged or deleted.
- The CMS must create 'stub' pages (empty pages with headers/footers) for each WWW MatrixMaxx module/section that will be linked and used on the website. This is necessary so that the navigation status matches the location of the stub page.
    - Example: The association is using 3 different modules of MatrixMaxx-generated pages: the create/edit profile pages (create individual, edit individual, edit organization), meeting calendar and registration pages, and committee pages. These pages will probably be in 3 different sections (i.e., Profile, Meetings, and About), thus needing 3 different 'stub' pages that confirm styling, navigation, etc.

- - The stub pages need the following properties:
    - MatrixMaxx must be able to request them
    - There must be AT LEAST ONE default stub page, but most groups use a minimum of 2 stubs (a default stub and a meeting/event stub)
    - The content area of the stub page must be uniquely identifiable, and must be empty. MatrixMaxx prefers the content area to be a 'div' element with the CSS class 'content', but this can be customized as necessary.
    - The stub pages are NOT in the navigation. (These are NOT user-facing pages. These are 'fake' pages simply being created for use by the AMS code)
    - There should be some mechanism, documentation, or process to ensure that these special pages do not get deleted by the association staff at any point in the future.
- If the sites will be on different domains (e.g., www.domain.org vs. www2.domain.org), all relative URLs will need to be transformed into absolute URLs (e.g. /scripts/jquery.js becomes http://domain.org/scripts/jquery.js).
  - The MatrixMaxx AMS team assumes that this step will be undertaken by the CMS team.
- If there's a login form in the header or special content in the header for logged-in users (like a logout link instead of login form, or 'Hello, User!' message), the MatrixMaxx team will need to do some custom coding to make the draped pages behave like the pages in the main site. This is often easier if there are good selectors for those special elements (put them in div.loginForm container or something like that).
- If the design uses svg files, those files need to include the 'Access-Control-Allow-Origin' header to allow MatrixMaxx to execute them. Here's a relevant stack overflow discussion.
- Breadcrumbs may or may not need special attention from CMS or Maxx side; it's definitely something to check in testing.
- Page title may or may not need special attention from CMS or Maxx side; it's definitely something to check in testing.
- In terms of debugging, if the CMS is being developed and/or hosting with Matrix, and is on one of our dev servers, the debugging text at the bottom of the pages in our dev environment will include the stub path and its refresh data.
- And finally, the 3rd party CMS may need to open up any firewall restrictions that might block MatrixMaxx from pulling the 'stubs' that we need

Sample Sites

- SiteFinity CMS
  - http://www.fmi.org/
    - Sample MatrixMaxx-generated page that is being dynamically wrapped: Search For Company: FMI
- WordPress CMS

- - https://www.manufacturingcleaning.org/
    - Sample MatrixMaxx-generated page that is being dynamically wrapped: https://www2.manufacturingcleaning.org/forms/MeetingCalendar/
  - https://www.awpa.org/
    - Sample MatrixMaxx-generated page that is being dynamically wrapped: https://www.awpa.org/forms/MeetingCalendar/
  - http://www.unitedfresh.org/
    - Sample MatrixMaxx-generated page that is being dynamically wrapped: http://www2.unitedfresh.org/forms/store/CommercePlusFormPublic/search?action=Feature
- MURA CMS
  - https://www.naruc.org/
    - Sample MatrixMaxx-generated page that is being dynamically wrapped: https://maxxwww.naruc.org/forms/CompanyFormPublic/viewCommissionRoster?id=764000007F1
    -