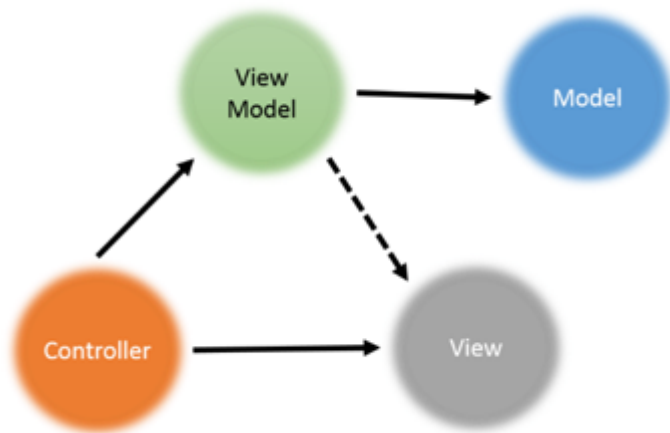


View

[Go To Naming Conventions](#)

For information regarding razor views, click the images below



Responsible for presenting viewmodels, viewbags and other data structures in the format of the end user requirements

- Should mainly contain presentational code, such as HTML, to format and render data
- No code that performs explicit DB queries may be added to the view and must rather be placed in the view model.
- Should avoid direct access to `$_GET`, `$_POST`, or other similar variables that represent the end user request.
- May access properties and methods of controllers and view models directly only for presentational reasons

Reusing Views

Layouts

Common presentational areas (e.g. page header, footer) can be put in a layout view.

TODO - add link to Layout page in The Standards.

Partial Views

Views without a layout that reuse fragments of presentational code e.g. used in modals.

- Must be used if only parts of a page with data must be refreshed e.g. List with paging
- Represents code that can be used by multiple views in the Area.
- A partial must always start with an **underscore**.
- If the partial is only used inside the specific controller it must be in that specific folder. E.g. Details for LeaveGroupController will be in the LeaveGroup folder with the name: **_Details.cshtml**.
- A partial that shows items linked to the specific entity, must have the name of the collection linked to the entity, and must be in the folder of the entity. E.g. Leave Types linked to Leave Groups will be: **_LeaveTypes.cshtml** and it will be in the LeaveGroup folder.
- A partial that contains a list of items that can be linked to the entity must follow the convention in 3. and must follow these naming conventions:
 1. If it is a multiple link list (with the delink, link and in use buttons): **_LeaveTypesLinkMultiple.cshtml**
 2. If it is a picker list that populates a textbox on the parent page (you can only link one item at a time): **_LeaveTypesSelectSingle.cshtml**
- A partial with a list of items that is generic and shared over multiple controllers must be in the Shared folder.
 1. If it is shared only in the current Area, it must be in the Area's shared folder: **App/Areas/Module/Views/Shared**
 2. If it is shared over multiple areas it should be in the global shared folder: **App/Views/Shared**
- A partial that is used for navigation should follow these naming conventions: **_LeaveTypesSideNav.cshtml** is a partial for Leave Groups that is used to navigate between Leave Types linked to a Leave Group
- Exceptions:
 1. When using a partial in a one view only but for AJAX posting i.e. an action returns **PartialViewResult** in controller
- If the partial is shared between views, the javascript needs to reside on the partial itself.
- Javascript shared among multiple partials with the same parent view should reside on the parent view.

Helpers Classes

In views we often need some code snippets to do tiny tasks such as formatting data or generating HTML tags.

- Small segments can be placed directly in the view, but larger segments must be moved to a class file
- An example of a helper and its use in a view

The method in the helper class

```

@helper StepLink(string name, string icon, string actionText, string actionIcon, string onclick = null, string href = null, string target = null) {
    // Must have action text as well as an onclick or an href
    var clickable = !String.IsNullOrEmpty(actionText) && (!String.IsNullOrEmpty onclick) || !String.IsNullOrEmpty(href);

    <div class="step-link">
        <div class="step-link_name">
            @System.Web.Mvc.MvcHtmlString.Create(name)
        </div>

        <a class="step-link_icon" onclick="@onclick" href="@href" target="@target" @if (!clickable) {<text>style="cursor: default"</text>}>
            <i class="fa fa-@icon"></i>
        </a>

        @if (clickable)
        {
            <a class="step-link_action primary-bg" onclick="@onclick" href="@href" target="@target">
                <i class="fa fa-@actionIcon"></i> <span>@actionText</span>
            </a>
        }
    </div>
}

```

The implementation of the class and method in the view

```

<div class="text-center">
    @PathwayHelpers.StepLink(Model.QuestionMarkAssessment.AssessmentId, "question-circle", "Launch", "paper-plane", Model.Url)
</div>

```

TODO: add examples from v8 mvc page

Ensure list data bound during post (use for instead of for each)

Difference between Html.Form and Ajax.Form

Use of hidden field variables and where they should be placed in a view

Revision #11

Created 17 September 2020 01:55:44 by Theuns Pretorius

Updated 8 October 2020 05:36:17 by Theuns Pretorius