

How to create a new user license type

- All changes are made in the Utopia solution in the Utopia git repository
- See this [PR for the Compliance License](#) for a previously implemented example

C#

UtopiaSharedClasses\Enums\UserLicenseEnums.cs

- Add a value to the `UserLicenseEnum` enum
- Add a value to `AllowedNodeTypesByUserLicenseType` dictionary (if applicable)

UtopiaSharedResources\Utilities\UserLicenseUtilities.cs

- Instantiate a new `UserLicense` in the `UserLicenses` static method

UtopiaSharedClasses\Enums\AccountFeatureEnum.cs

- Add a value to the `AccountFeatureEnum` enum
- Add a value to the `UserLicenseAccountFeatures` array. The licenses in the UI will display in this order.
- Add a value to the `MeteredAccountFeatures` array (assuming that the license will be metered)
- Add a value to these other arrays (if applicable)
 - `UserLicenseAssignmentFallbackOrder`
 - `RoleLicensesWithConcurrentUserSupport`
 - `AdvancedOCRAccountFeatureTypes`
 - `MeteredOCRAccountFeatureTypes`
 - `UsageFeatureTypes`

UtopiaSharedResources\Language\EnumDisplayNames.resx

- Add a resource value for the title of the license
 - The name must be `AccountFeatureEnum_{name-of-account-feature-enum}`, e.g. `AccountFeatureEnum_ComplianceUserLicense`

UtopiaSharedClasses\Classes\AccountClasses.cs

- Add an `int` property to the `AccountUsage` class

BusinessLogic\Services\AccountService.cs

- In the `GetAccountReportAsync` method, update the assignment to the `userCountLookup` dictionary to also add the counts of the new license type

BusinessLogic\Services\AccountUsageService.cs

- In the `GetUsageAsync` method, add a line to set the count of the new license

DataAccess\Methods\RoleMethods.cs

- In the `GetUserLicenseCountLookupAsync` method, add a line to set the count of the new license

JavaScript

wwwroot\Client\app\accountFeature\accountFeatureServices.js

- In the `determineFeatureUsageMessage` function, add a `switch case` for the new license

wwwroot\Client\app\auth\authFactories.js

- In the `getUserRoleOnAccount` function, add a block that returns a role for the new license if one exists

wwwroot\Client\app\permission\permissionServices.js

- In the `getPermissionPresets` function, add a block that sets the permission presets of the new license

wwwroot\Client\app\roles\roleServices.js

- In the `isUserLicenseType` function, add the new license to the `OR` clause
- Add a function called `is{name-of-license-type}UserLicenseType`

Revision #7

Created 31 October 2022 18:20:18 by Joe DeStazio

Updated 17 April 2023 17:29:19 by Joe DeStazio