# ATS – Automated Test System

## What is implemented so far

### Folders Management

* Sub-Folders tab – Displays the sub-folders of the folder. Double-Click on s sub-folder keeps drilling down.
* Properties tab – Folder’s properties.
* Content tab – This is empty now. When we will have Content Management, we will drag content to this tab, essentially turning the Folder to a Project.
* Certificates tab – This is empty now. When we will have Certificates, we will drag them to this tab.
* Notes – A placeholder for developing the Notes module.

### Tree

* Add \ Delete Folder. (No need for Edit; standing on a node == Edit).
* Cut \ Copy \ Paste.
* Refresh.
* Drag and Drop of folders in the hierarchy.

### External Modules

* Content Management – Placeholder for developing the Content Management interface and presenting the results in a tree.
* Certificates – Placeholder for developing the Certificates interface and presenting the results in a list.
* Search – Currently searching on Name and\or Description and\or User.

### Few words on Search

When you get into the program and simply want to search on Name, fill the search string and hit ‘Find Next’. Keep on hitting it till you get to the folder you want.

If you want more advanced search, hit the ‘Search’ button. This will open a control for you to fill the search string and specify where to look for. When you are done and hit ‘Search’ from the control, you will now be searching on the selected fields specified whenever hitting ‘Find Next’ button.

The search is NOT case-sensitive (meaning it will locate ELI, Eli, etc.). Also, it is not looking for whole words or beginning of words – whenever it sees the string, the search is successful.

## Remaining Tasks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Description | Person | Days | Comments |
| 01 | Content Management window |  |  | Without Drag & Drop |
| 02 | D&D from CM window to Folders | Eli |  |  |
| 03 | Certificates window |  |  | Without Drag & Drop |
| 04 | D&D from Certificates to Folders | Eli |  |  |
| 05 | Activate Content | Eli |  | Including files collection |
| 06 | Notes module |  |  |  |
| 07 | Advanced Search | Eli |  | Search Project Code, Notes, Content |
| 08 | Projects Management Phase 1 |  |  | Basic (similar to Folders) |
| 09 | Projects Management Phase 2 |  |  | Additional Features (NOT Groups) - such as Disable\Enable etc. |
| 10 | Projects Management Phase 3 |  |  | Versions |
| 11 | Projects Management Phase 4 |  |  | Groups (Multi-Makat) |
| 12 | Projects Management Phase 5 |  |  | Presenting Content in Projects – Need to change the DD; this is a CR |
| 13 |  |  |  |  |
| 14 |  |  |  |  |
| 15 |  |  |  |  |

## Notes to Developers

* All the activities you will be required to do were already implemented for Folders. Make sure you understand how things are working and where they are located.
* The above is especially important for:
  + How to implement Models, Views, and View-Models; where to place them.
  + The way to interact with the Database via code in the BLL (Business Logic Layer).
  + How to handle TreeView (whereby every node is a mini View-Model).
  + Implementing simple validation (via attributes in the VM) and more complex ones, if needed, in the BLL.
* I have already prepared skeletons Views and View-Models for Notes, Content, Certificates, Projects, and Versions. Just put logic into them as per the DD.
* As you recall, I already sent you the instructions on how to develop Certificates and Content Management. I am adding these instructions again here.
* For every activity \ context-menu you are adding, check in the Detailed Design if the activity is privileged. If so, follow the logic I have implemented (for example in the ‘Add Folder’) and make sure you do the same.

## Development Instructions – Content Management

* In the Flyout we should have a TreeView that will present the data we receive via the interface with SCD’s Content Management. Once this TreeView is shown, the user will be able to perform Drag from a TreeView node onto the content tab of the Folder\Project.
* I have already created the entire mechanism, as shown above. All the messages that flow from the Folder’s ViewModel (VM) to the MainWindow VM (that tell it to open the Flyout and which control to present there) are done.
* You need to develop the mock to Content Management that will return to you a Dictionary (or any other collection data type, as agreed with Alex). Once you get this, you will populate a TreeView.
* Make sure you see how I did the TreeView of the Explorer. You need to do the same. A TreeView in MVVM is a hierarchical collection of VMs, each entry in the collection is a VM and a View.
* You need to do all your work inside the User Control I prepared (ContentManagementView and ContentManagementViewModel), both located in the ExplorerModule project.
* The Mock itself should be developed outside of our solution, and the compiled dll should be added as a reference to the ExplorerModule project. When we receive from Alex his final dll, we will drop the reference to the mock and include a reference to Alex’s dll.
* Make sure the data you generate in the mock has enough content and hierarchy levels to simulate the live data we will get.
* Once everything is created, we will add the Drag and Drop capabilities between the Content TreeView and the target ListView in the Folder (which is not created yet).

## Development Instructions – Certificates

* Similar to the above, only the interface with SCD is not defined yet and should be clear. The data returned should be flat, and therefore should be presented in a ListView.
* The View is called CertificatesView, and there is a respective VM.