FileOrganizer – Planning

# Key Features:

1. Create Files/Folders with custom filetype with naming pattern
   1. Build the custom name by adding “blocks”
   2. One “block” can be:
      1. A custom string
      2. A number with up to 3 leading zeroes
      3. (the original name?)
2. Delete Files/Folders by naming pattern
   1. Examples:
      1. Delete everything that ends with a “\_test”
      2. Delete every jpg
      3. No delete by time
3. Easy to use GUI
   1. Question marks with help pop ups/hints?
   2. Regions with panels (root path selection are, building the naming pattern are)
      1. CSS: gradients in BG? Color?
   3. Animations that slide the view to the clicked area
   4. Show the current name building blocks as drawn boxes or in a locked text field
   5. Show an example of how the finished name for the files will look like

# Bonus Features:

1. Create folder structure by text file (own syntax = .fml)
2. Create context menu entry in windows explorer so that a user can start the program with the current folder as the root path
3. Create a “Show folder in Explorer” button so that a user can see the created files without navigating to the folder him-/herself

# Requirements:

* Class diagram (yED desktop)
* 1 x Inheritance
* 1 x Association (UML)
* min. 4 classes
* 3 x Loops
* Comments 🡪 JavaDoc (do comments last, JavaDoc while writing as it is collapsible)

# Tasks / Roadmap:

1. Learn about Java Doc
2. Build working GUI in FXML and wire it up with a controller
   1. Implement Material Design (or one style) with CSS? In regards to color and slight shadows behind objects
   2. Animation possible for switching between the modes? (see drawn GUI-Diagram for more details)
   3. Use a nice font (google fonts?)
   4. Use nice icons (google material icons?)
   5. Use color
   6. Question marks with help pop ups/hints?
   7. Regions with panels (root path selection area, building the naming pattern area)
      1. CSS: gradients in BG? Color?
   8. Show the current name building blocks as drawn boxes or in a locked text field
   9. Show an example of how the finished name for the files will look like
   10. Vertical Labels for the area (or labels at the top center and vertical labels for “go back to main menu”)
3. Implement simple File-Creation-Service
   1. Locked Name & FileType in a specified (root-)path  
      Build custom name and FileType together with NamingService
4. Implement simple Folder-Creation-Service
   1. Locked FolderName  
      Build custom name together with NamingService
5. Implement Naming Feature Service for Files and Folders
   1. This should be usable by both services
   2. It is only about building a name, adding the FileType should be outside of this service or as the last step
   3. Check, that the name is valid for files/folders (e.g. no ! “ ? & % $ etc.)
6. Polish/Expand the File-Creation-Service
7. Polish/Expand the Folder-Creation-Service
8. Implement simple File-Deletion-Service
9. Implement simple Folder-Deletion-Service
10. Polish/Expand the File-Deletion-Service
11. Polish/Expand the Folder-Deletion-Service
12. Plan the FML (Folder Modeling Language)
    1. Folder Structure in Text Form:
    2. # or // as comment
    3. -FOLDER\_NAME
    4. --SUBFOLDER\_NAME
    5. ---SUBFOLDER\_NAME
13. Implement the Create-Folder-by-File-Service
    1. Check for correct Foldername
    2. Preview the Folder-Structure in a tree view/or something else
14. Final Polishes
15. Bonus Features
16. DEV LOCK:
    1. Draw class diagram
    2. Add comments to f\*cking everything
    3. Enhance JavaDoc
    4. Check the program for given requirements (loops, classes, etc)