Analysis and Design Documentation

- GUI Design Specs -

** Functionality **

Basic
- load group of image files
- tile images within specified constraints (i.e. in a square with edge size as a power of 2)
- export the tiled images as one image file
- export an XML file with data for each individual image's name, dimensions and location on the assembled sheet

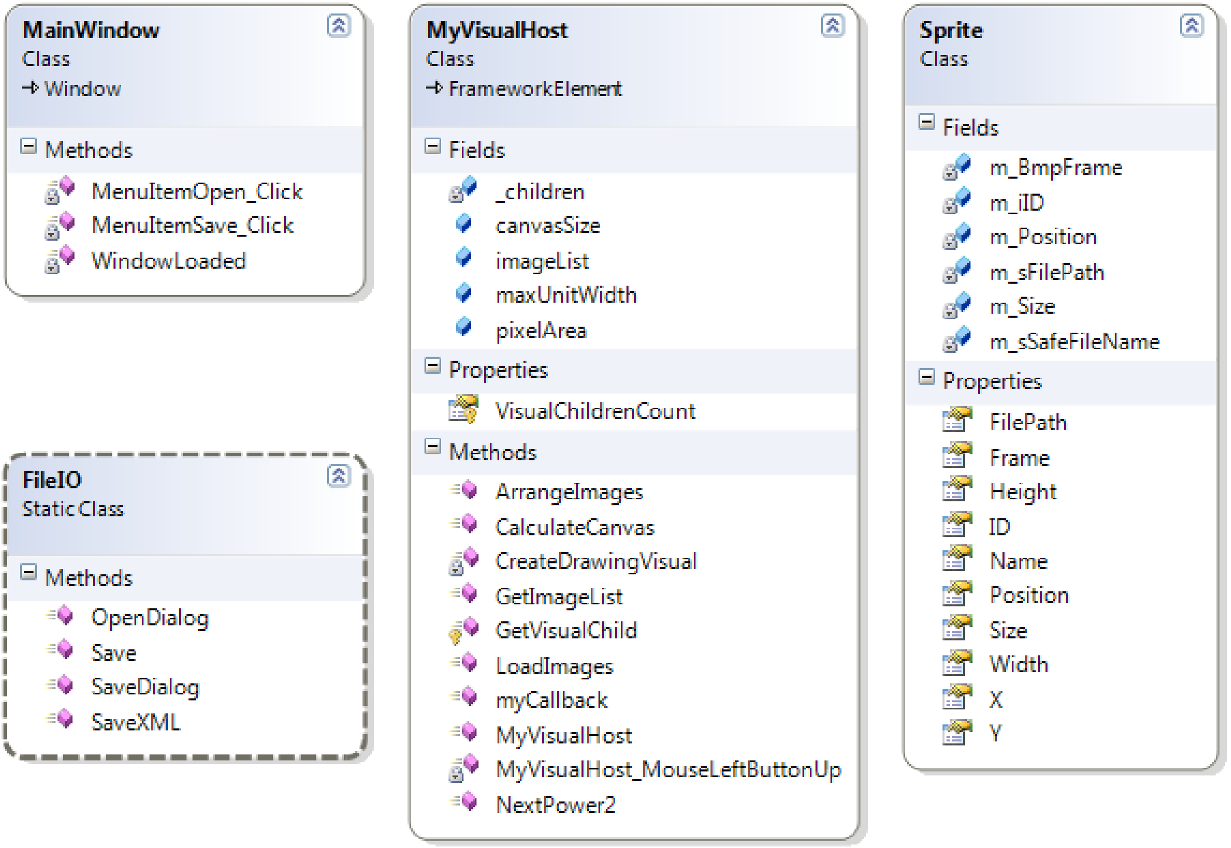
Desired
- select a group of images within the preview and create an animation
- export an XML file with animation data
- customize XML element and attribute names

** UI **

Basic
- save button to export assembled sheet and XML file (Open file dialog runs automatically at launch)
- preview area to display arrangement of images
- appropriate Open and Save dialog options (file formats, defaults)

Desired
- load button
- - allow folder selection
- - allow asynchronous loading?
- preview window for animation
- info window for sheet or individual image data
- make it look gooder

Class Diagram



Data Structures

MyVisualHost Class
-derived from FrameworkElement

Fields:

- _children……….VisualCollection = collection to hold DrawingVisual objects
- imageList………master List of Sprite objects
- pixelArea……….the total area of all images (in pixels)
- maxUnitWidth….greatest width of an image *won’t be used after changing sort algo.
- canvasSize…….size of canvas side (square)

Sprite Class

Properties/Fields:

- FilePath………...the full file path of the image
- Name…………...just the file name and extension
- ID………………..a number, equal to index in master List of Sprites
- Frame…………..a BitmapFrame object for the actual image
- Position…………vector representing location relative to the Canvas
- Size……………..represents image dimensions

Algorithm Specifications

** **Load Images** **

- run Open File dialog and get the paths and names for all files
- initialize Sprite objects and populate master Sprite List

- Determine Canvas Size:

(current implementation makes a square with power of 2 edges)

- find the square root of the total pixel area of the images
- find the next greatest number that is a power of two
- set the canvas properties to new desired dimensions

- Sprite Arrangement

```
- create "cursor" for image insertion point
- - there are separate X and Y cursor values for determining the unit's
    placement on the canvas (in pixels)
- for each image in the list:
- - if it is the first image, simply set it's position to current cursor value (0,0)
- - otherwise:
- - - if the unit will not fit on the current row:
- - - - move cursor.Y down to the bottom of unit at the beginning of the row
- - - - move cursor.X back to 0
- - - - set unit's position
- - - otherwise:
- - - - add previous unit's width to cursor.X for proper offset amount
- - - - set unit's position
```

- create a DrawingVisual for the image and add it to the collection