Group 2 Modules Description

# Text Module

## Input Attributes (derived from PWS)

Required;

* Int xStart
* Int yStart
* Int xEnd
* Int yEnd

Optional;

* String font
* String fontColour
* String lineColour
* Int fontSize
* Int startTime
* Int duration
* Int layer

Sub-elements (optional for specifying typefaces);

* Boolean bold
* Boolean italics
* Boolean underlined
* Boolean highlighted
* Boolean superscript
* Boolean subscript
* Int orientation

## Outputs

A JPanel fitted to the size of the text string. The size is derived using the xStart, yStart, xEnd and yEnd values that are required for all text objects.

A JPanel containing the rendered text object will be returned per text object. The returned JPanel will need to be placed at the correct location on-screen, which will require implementation by individual teams. This is due to different teams using different portions or areas of the screen, and a provided implementation would likely be incorrect for others.

Multiple JPanels should be able to be displayed on-screen at once. This will make use of the layer tag to define in what order they should be visible to the viewer.

## Additional Functionality

* Ability to render superscript
* Ability to render subscript
* Ability to render characters from Greek alphabet
* Ability to render a limited range of common mathematical operators and symbols
* Ability to highlight portions of the text
* Ability to set the orientation of text object within the JPanel

# Image Module

## Input Attributes (derived from PWS)

Required;

* String urlName
* Int xStart
* Int yStart

Optional;

* Int width
* Int height
* Int startTime
* Int duration
* Int layer
* Int branch
* Int orientation

Accepted File Formats;

* PNG
* JPG
* GIF

## Outputs

A JPanel fitted around the size of the image. If the height and width tags are specified, these will be used to define the size of the JPanel. Otherwise the size will be derived from the image itself.

A JPanel containing a single rendered image will be returned. The returned JPanel will need to be placed at the correct location on-screen, which will require implementation by individual teams. This is due to different teams using different portions or areas of the screen, and a provided implementation would likely be incorrect for others.

Multiple JPanels should be able to be displayed on-screen at once. This will make use of the layer tag to define in what order they should be visible to the viewer.

## Additional Functionality

* Ability to set the orientation of the image object within the JPanel

**Video Module Function**

Play .avi and .mp4 files

Set position

Set start time

Set orientation

Full screen

Play, pause, stop

Slider bar control thing (Can’t for the life of me think what this is actually called)