



Iteration 1 Report

Steve Thorpe

1. Aims

- All handlers (video, graphics, text, images and audio) to be completed and compatible with PWS
- XML parser to be completed to PWS with a completed holding class for information extracted from the XML file.
- Local access via a file browser to an xml playlist.
- Basic logic for creating and displaying the slideshow using some of the handlers to have been implemented.

2. Work Completed

Text and images handlers have been completed to PWS with the exception of branching, layering and the bold italic and underline elements of the text handler. The video handler coding has just begun in sprint 3, which started on 11/03/14, with a completion deadline of week 3 of the Easter holiday.

The audio module has been delivered by the contracted group; it requires some minor modification to make it compatible with our intended slideshow creation structure. The threads which control the start time and the duration the audio object exists for are currently called outside of the audio handler class, they must be moved inside the class in order to operate in time with the text and image modules. Like the text and image module it also requires layering and branching to be added to it.

The graphics module has been contracted to be delivered by week 2 of the summer term, the video module will be exchanged for this at the same time.

The XML parser with the holding class conformed to PWS standard until the PWS was changed on 10/03/14. The most significant change has been the bold italic and underlining elements of the text objects being changed to attributes. This does not require a significant amount of work to account for; it is simply an extension of the existing XML logic and has been a beneficial change for the implementation of bold underlined and italicised text in the text handler.

The local access of an XML file has commenced in sprint 3, the pair responsible for it believe it will be completed well before the sprint 3 deadline.

There had been no logic implemented into the program for slideshow generation before 12/03/14.

3. Software Manager's Comments

All team members have worked well within the group and the code produced has been of a high quality. Improvements were required to the commenting of the code however after conducting a tutorial with the development manager on using JIRA and commenting practice, the standard of the commenting improved.

The coding has not progressed as far as intended for several reasons. The principle reason I believe is our inexperience in predicting how long it will take to produce the

code, it has taken much longer than our original estimates to produce each user story. The delays also been driven by the change from using Java Swing to JavaFX. This was a necessary change in order to sell modules but our unfamiliarity with the classes in JavaFX has exacerbated the delays that we have experienced. As coding progresses all of the group members are becoming more fluent with JavaFX and are producing code faster. Ultimately I believe the change for JavaFX will be very beneficial to the product: the GUI options are superior to Swing; it has several classes that may be useful for producing the slideshow itself that are unavailable in Swing; and it will extend the lifetime of the product as Swing is to be discontinued in favour of JavaFX.

The first integration of the product took significantly longer than expected after already being delayed by producing the Image handler by week 9 for group 4, which required the efforts of the entire software team and another group member. We aim to make up this time however as much of the code produced for the image handler can be used for the other handlers, all of their code will be essentially the same it will just output different objects. The other integrations will also take less time as we will be integrating single elements into the existing product as opposed to bringing several elements into a new product.

Despite these set-backs however the work velocity is increasing and the product is expected to be completed on time with all of the features planned in the functional specification. The deadlines for the remaining sprints are:

- Sprint 3: End of Easter week 3
- Sprint 4: End of Summer week 1
- Sprint 5: End of Summer week 3

This timeline then allows for four weeks of integration, testing, bug fixing and GUI finalisation.

