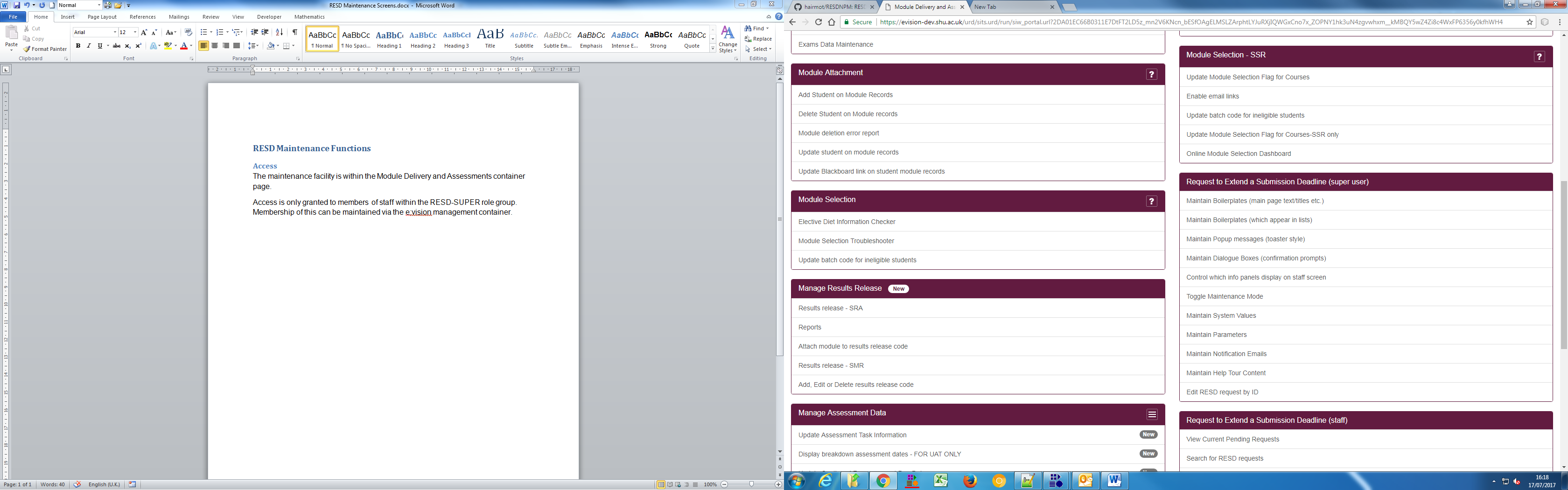
# RESD Maintenance Functions

## Access

The maintenance facility is within the Module Delivery and Assessments container page.

Access is only granted to members of staff within the RESD-SUPER role group. Membership of this can be maintained via the e:vision management container.

## Overview of functions



Most of the functions are ultimately manipulating the data within either UDD or SLP.

### SLPs:

Maintain Boilerplates (main page text/titles etc.)

Toggle Maintenance Mode

Maintain Parameters

### UDDs

Maintain Boilerplates (which appear in lists)

Maintain Popup messages (toaster style)

Maintain Dialogue Boxes (confirmation prompts)

Control which info panels display on staff screen

Maintain System Values

Maintain Notification Emails

Maintain Help Tour Content

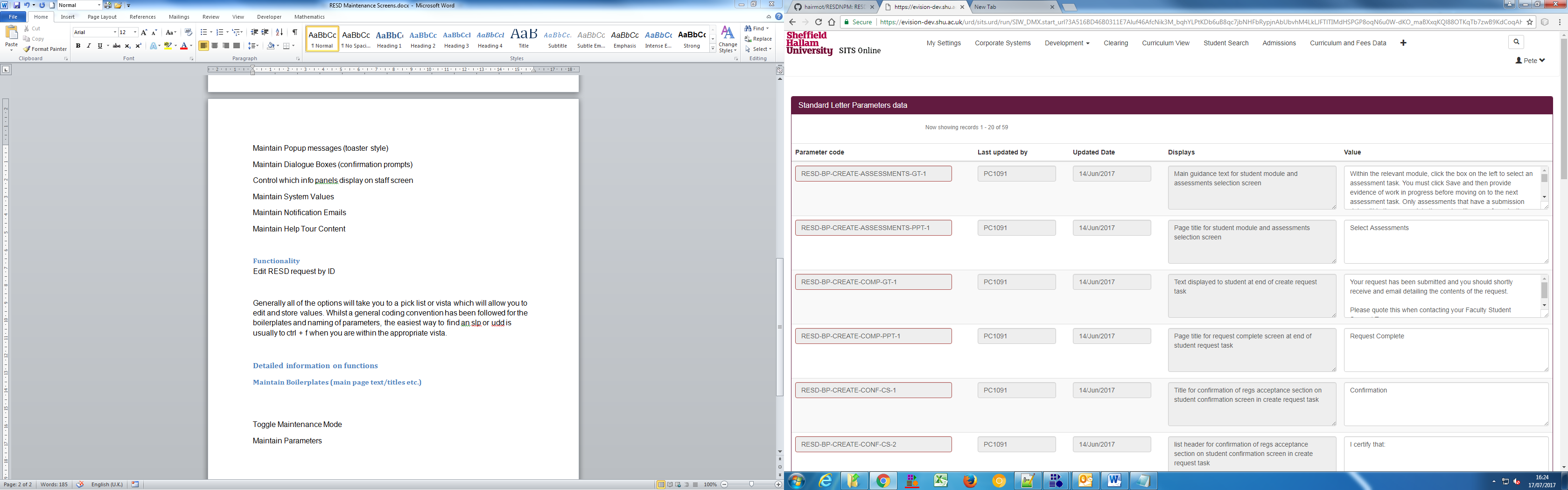
### Functionality

Edit RESD request by ID

Generally all of the options will take you to a pick list or vista which will allow you to edit and store values. Whilst a general coding convention has been followed for the boilerplates and naming of parameters, the easiest way to find an slp or udd is usually to ctrl + f when you are within the appropriate vista.

## Detailed information on functions

### Maintain Boilerplates (main page text/titles etc.)

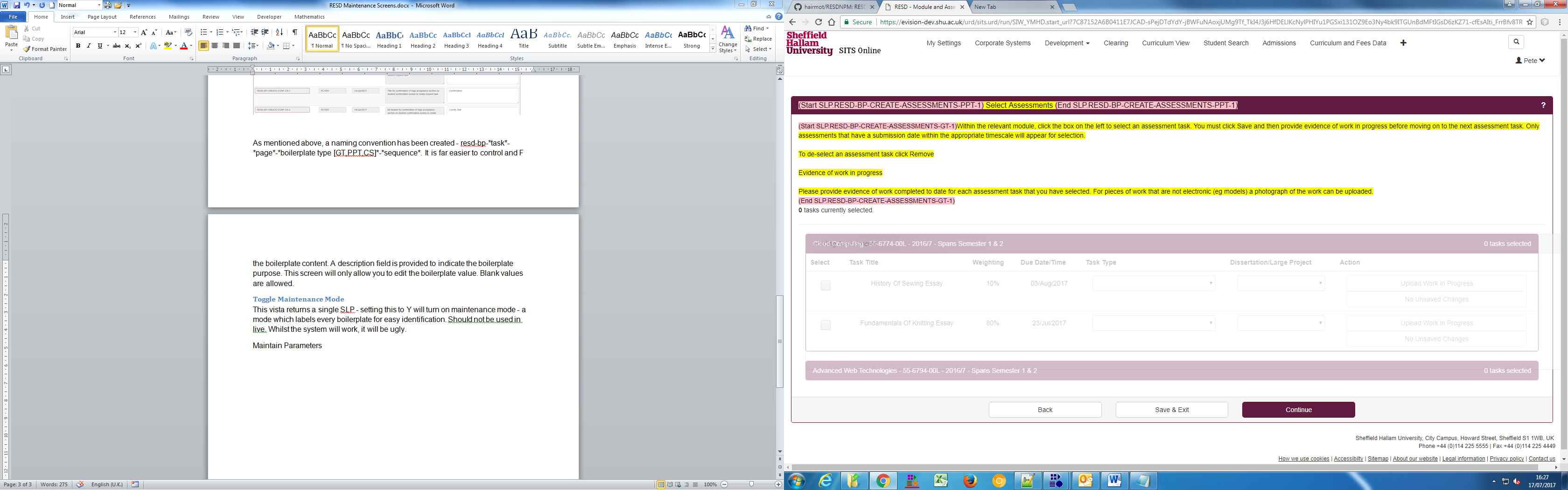


As mentioned above, a naming convention has been created - resd-bp-\*task\*-\*page\*-\*boilerplate type [GT,PPT,CS]\*-\*sequence\*. It is far easier to control and F the boilerplate content. A description field is provided to indicate the boilerplate purpose. This screen will only allow you to edit the boilerplate value. Blank values are allowed.

### Toggle Maintenance Mode

This vista returns a single SLP - setting this to Y will turn on maintenance mode - a mode which labels every boilerplate for easy identification. Should not be used in live. Whilst the system will work, it will be ugly.

Example:



### Maintain Parameters

This vista has bundled together a few SLPs (defined in the validation field of the first DMF in " RESD-MAINTAIN-R" DML).

task eligibility latest date: the latest you are allowed to include an assessment in a request. Takes a number e.g. 5. minus numbers will open up the process to after the assessment date.

task eligibility earliest date - how far before a due date can an assessment be included.

Max length of summary provided on summary of circumstances screen. initiallty set to 300 - will update the validation javascript.

Maximim length for explanation of why no evidence provided on summary screen - same as above.

### Maintain Boilerplates (which appear in lists)

There are a few boilerplates which include a list. As lists require special html markup, these have been stored in lists. This allows you to add and remove records and the markup will be nicely formatted. Examples include the confirmation bullet points and eligibility criteria in the create task.

### Maintain Popup messages (toaster style)

These are the pop up notifications that appear in the top center of the screen and fade away. The best way to find these is to ctrl + f the message.

### Maintain Dialogue Boxes (confirmation prompts)

These are the prompt boxes which almost always contain an ok or cancel button.

It's easier to find these by ctrl + f. In addition to the message content, you can change the title of the prompt box.

### Control which info panels display on staff screen

This is specifically for the RESD-PROCESS task. When the member of staff is inputting a decision, relevant info is displayed at the top of the screen. We included the function to disable some of these panels if it were to be decided later down the line that the info is no longer useful.

### Maintain System Values

These are UDD lists which are referenced throughout the process e.g. outcomes, valid task types.

These parameters will accept slp or SRL content (in case the logic needs to be expanded in future)

### Maintain Notification Emails

This is not a text editing feature, just a place where the content can be input. I would suggest designing emails outside of SITS and importing them.

### Maintain Help Tour Content

These are the info boxes displayed when a user clicks ?

These can be added to by creating an SLP with code <<TaskName>>-<<TaskStep>> and a sequence of (n+1)\*10. The box is attached to an element on screen by using a css selector.

If this still does not work, you may need to include the SLP @RESD-TOUR-OBJECT - this generates a JSON object for the javascript to pick up.