This is a top down design where I have used stepwise refinement to break up the assignment smaller and easier to create, design and manage sections. The user interface is broken down into four parts. Get user input, read file where the file is designated by the user, use TextLocaliser() to convert all date and currency values in the file and create and write to a file with the same content as the input file but the date and currency values have been localised. The user input includes, input country, input file name, input file location, output country, output file name and output file location.

The TextLocaliser() must implement an interface given by the assignment brief. The interface must include the following functions, Localise(), LocaliseDate(), LoadDateFormats(), LocaliseCurrency() and LoadCurrencyFormats(). The Localise() function looks through the input text and find the dates and currencies in the text and use LocaliseDate and LocaliseCurrency to convert them to localised values and return the values. The LocaliseDate() function uses LoadDateFormat() to load all the hard-coded elements into a map that is then used to reassign the date fields. It also must find the delimiter of the date for example the delimiter of “24/02/2012” would be “/” as it is used to space the date fields. This function also must change the year value to two or four digits depending on the output country and finally this function must rearrange the input date fields into correct order. This can be done by assigning the input date field to the input date format and calling them in the order of the output date format. Then return the value to Localise(). The LocaliseCurrency() function uses LoadCurrencyFormat() to load all the hard-coded elements into a map that is then used to see if the output country need “,” or “.” in the currency and to see what symbol they use and what location the symbol is on the currency. This function also must find and multiply the input value by the correct exchange rate, add in the “,” or “.” Depending on the output country and add in the symbol in the correct location on the values. Then return the value to Localise(). LoadDateFormat() is passed a pre-created map then add the hard-coded values to the map and then returns it to LocaliseDate(). LoadCurrencyFormat() is passed a pre-created map then add the hard-coded values to the map and then returns it to LocaliseCurrency().