**Assessment Feedback**

|  |  |  |  |
| --- | --- | --- | --- |
| **Student Name:** | Lance Flatman | | |
| **Assessment Code & Name:** | 33130/01 Develop a UI to interact with a database | | |
| **Course Code & Title:** | ICT40515 Certificate IV in Programming  Units:  ICTPRG403 Develop data-driven applications  ICTICT420 Develop client user interface  ICTPRG410 Build a user interface | | |
| **Assessor:** | Justin Mclean | | |
| **Date of Final Result:** | 7/05/2019 | **Final Result:** | Distinction (DN) |
| **General / Opportunities for improvement:** | Hi Lance,  Excellent work here, and thanks for addressing Damian’s feedback. There’s some room for minor improvements.  Some general advice on your code:   * Try not to leave commented out code in you code, you’re using version control so just remove it. * Tight loops i.e. a loop with nothing in the body (even if they are not being used) are usually not a good idea. * Take care and try not to use hard coded values in your code e.g SelectedItems[0].SubItems[1] * Take care not to repeat code and use function, even small one-line functions can make code more readable. * It would be good to see some more comments explaining why you are doing things in a certain way. * Putting state into a single global class isn’t that scalable, I know it’s suggested to do it that way, but you may want to investigate alternatives or break it up a little. * Take care in selection of variable names, try not to abbreviate them too much. * With modern IDEs and particularly when using a strongly typed language there no much point in using Hungarian notion in naming variables. Opinions may vary on this, but his old joke does have some merit I think “vUsing adjHungarian nnotation vmakes nreading ncode adjdifficult”. Anyway think about it and come up with your own view on it. * Try not to use transactions where they are not needed. A single SQL statement is atomic and there’s no need for a transaction. You only need to use transactions were you have multiple SQL statements and the second one depends on information from the first. Transactions are almost never needed for stored procedures (by design) or select statements.   Some general advice on your UI:   * There is probably no need for : after tables especially in titles, buttons and menus. * There little need to ask a user are they sure when they want to add something, you usually do this when they want to do something that can’t be undone e.g. deleting something. | | |
| **What I really *liked* about your assessment:** | Easy to understand code with consistent naming and well-chosen methods and variable names. A list of clearly explained, comprehensive test cases. No unexpected application crashes. | | |