**Topic 3: Pseudocode and Documentation**

**Formative Assessment 03**

**Pseudocode and Documentation (115388)**

*Hello and welcome to the assessment. Here you’ll prove to the world just how much you know and understand about what you’ve just learnt in the learner guides. This is an important part of your time at Umuzi because once this is done, you’ll be certified! So please, take this time to learn everything you can! Take a look at some pointers below with regard to answering the questions…*

* *Be specific*
* *Write professionally - no shorthand!*
* *Your answers must be original and come from your brain and your brain only.*
* *No copy/paste tricks! Our markers have seen it all and will know if you’re taking shortcuts.*
* *Remember, sloppy or poor work will be sent back to you to do again, so do it properly the first time and you’ll be done in no time.*
* *Ask for help at any time. Ask your friends, a manager, anybody!!*
* *Don’t skip any questions! You must do them all!*
* *You’ll see two boxes after each question - one for your answer and one for the marker’s comments. DO NOT delete the marker’s comments if you are required to resubmit your work after the first attempt. Should you have to do it again you will see a new box* ***under*** *the marker’s comments, so fill that one out in* ***BLUE****. Remember!! It’s not the end of the world if you have to resubmit. You’re here to learn, so don’t beat yourself up if you don’t get it right on the first go. Obviously, try your best to get it right on the first attempt, but if not, you have another chance to do it properly!*

*Ok, and that’s that! Time to get to it! Good luck, have fun and enjoy! :)*

**Enter your name and surname below**

|  |
| --- |
| **Luvo Spofana** |

**1.** **Describe four best practices for Agile software documentation [8 Marks](8 - SO:1 AC:1-2)**

**Your answer below**

|  |
| --- |
| **Only the relevant information – Agile suggests that only relevant, necessary and non-obvious information should be documented. Customer documents should contain information that customers require and will be useful to them.**  **Wait before documenting – It’s best to document late, when there is no way that the implemented decision can be turned and undone, this helps keep your information accurate and true, and saves a lot time and money that would’ve been spent on redoing the documentation.**  **Be specific – Documentation should not be general and vague, it should be specific to each project because each project has its own uniqueness so making use of the same documentation template for different projects might not be ideal because some might have unnecessary fields while other are missing important content fields.**  **Keep it all in one place – Documentation of technical designs and developments should be orderly kept and easy to locate and access, it is important to keep related documents in one place (web/wiki page with multiple sections) or folder ( GitHub repository that is accessible to everyone)** |

**Marker’s Comments**

|  |
| --- |
|  |

**2. Explain how and why to document each function in a program you develop [5 Marks](5 - SO:2 AC:1-3)**

**Your answer below**

|  |
| --- |
| For every program you develop, it is important to document each function because if the code is open-source or distributed to end users, documenting it makes it easy for the users to understand the functions, especially when there are alternatives to the coded function, then users benefit from the documentation because they will understand the reason behind the chosen approach over the alternative. Functions are documented so that customers can feel more at ease as they make use of the program, and for keeping track of the project’s activities. The documented code can be used for audit purposes and new people in the team can use the documentation as training material as they may possibly contribute to the project.   * One documents each function by first writing the function’s pseudocode. * Then the pseudocode is converted into comments. * As the program changes, the code comments are updated accordingly. * To the documentation, you then add a summary of the function, along with high-level instructions. |

**Marker’s Comments**

|  |
| --- |
|  |

**3. Provide an example of how you would make a high-level summary of a software file you have created. [5 Marks](5 - SO:1 AC:1-4 & 378 SO:4 AC:1-4)**

**Your answer below**

|  |
| --- |
|  |

**Marker’s Comments**

|  |
| --- |
|  |

**4. Explain the importance of creating documentation for end users and reviewing that documentation for completeness, especially when developing a large program [6 Marks](6 - SO:3 AC:1-2)**

**Your answer below**

|  |
| --- |
|  |

**Marker’s Comments**

|  |
| --- |
|  |