Assessment Submission Coversheet:  
Introduction to C#

|  |  |
| --- | --- |
| **Student Name:** | Sarthak Saxena |
| **Student Number:** | S191509 |
| **Student Email** | S191509@students.aie.edu.au |
| **Course Stream:** | ICT50215 - Diploma of Digital and Interactive Games |
| **Assessment Name:** | Introduction to C# |
| **Units Covered:** | ICTPRG430 – Apply introductory object-oriented language skills  ICTPRG532 – Apply advanced object-oriented language skills |
| **Teacher/s:** | Jesse James Donlevy, Jay Yabsley |
| **Due Date:** | 8/30/2019 |
| **Date of Submission:** | *Will be automatically recorded on Canvas* |
| **Assessment Work Location** | Canvas/Drive location/file path |

**Declaration**

By submitting this work under my name, I declare that my submission is my own work with respect to plagiarism and do not violate any copyright laws. I have retained a copy of this assessment material that I can produce if requested.

Tick to acknowledge you have read and agree with this declaration.

Name: Sarthak Saxena Date: 9/3/2019

Assessment Submission Coversheet:  
Introduction to C#

**Work Submitted:***Tick to acknowledge you have submitted this part of the assessment.*

1. Application Design Document: The ADD displays the main concept of tool, the explanation of the flow of algorithms and data structures. The general idea, and a description of events and images of real time files of the tool.
2. C# Tool Application: The C# tool demonstrates a weapon being created, edited, saved and loaded. The C# tool can be used to sort and search weapons being used, through the use of multiple arrays.
3. Example File: The Example Files provided can be accessed to display loading of weapon files and save feature through editing weapon selected.
4. Tool Application Test and User Document: The Tool Application Test displays the tests performed to verify the C# tool as finished. The product contains a User document to be read to understand the process used in the C# tool through and through.
5. Remote Procedure Call Exercises: Two executables that when ran together, allow the client and server to communicate. The server and client need to be executed in an order to allow the communication to be established.
6. Data Structure and Algorithms Exercises: The submitted files include a double linked list and a binary tree. The double linked list and binary tree created in C#, allow the user to access the main features of each program.

*For more information on these parts, please go to the Subject Guide in the Introduction to C# section on* [*https://aie.instructure.com*](https://aie.instructure.com) *and open the* ***Subject Guide - Diploma of Digital and Interactive Games - AUS***

Name: Sarthak Saxena Date: 9/3/2019