

Open Awards

Quality Endorsed Unit



1 Unit Details

Unit Title:	Advanced Java for Android Games
Unit Reference Number:	CK3/4/WR/003
Level:	4
Credit Value:	9

2 Learning Outcomes and Criteria

Learning Outcome (The Learner will):	Assessment Criterion (The Learner can):
1. Understand how Java integrates with the Android Studio development platform	1.1 Describe the different panes of the Android Studio interface.
	1.2 Describe Activities and the Android Activity Lifecycle
	1.3 Write a simple Java programme using the Android Interface
	1.4 Run an Android project on both a virtual device and a real device.
2. Understand the concept of Java variables and operators	2.1 Implement Java methods, variables and operators
	2.2 Perform arithmetic and logical operations
	2.3 Devise the Java for loop including while loop, and do-while loop to produce: <ul style="list-style-type: none"> a) basic loops b) enhanced loops and c) nested loops
	2.4 Draw simple objects on the screen using the Canvas, Bitmap and Paint classes and the Android coordinates system
	2.5 Demonstrate the use of sound effects that can be incorporated into an Android project
3. Understand the concept of defining Java variables and objects which are	3.1 Create Java variables and objects which are

<p>automatically responsive to the screen size of the device.</p>	<p>automatically responsive to the screen size of the device.</p> <p>3.2 Demonstrate the use of both arithmetic and logical operators</p> <p>3.3 Draw text, gridlines, and rectangles</p> <p>3.4 Process the player's screen touches</p> <p>3.5 Create random numbers</p>
<p>4. Understand the concept of object-oriented programming</p>	<p>4.1 Use object-oriented programming and constructors</p> <p>4.2 Create polymorphic objects</p> <p>4.3 Implement inheritance</p> <p>4.4 Create own classes, interfaces and methods</p> <p>4.5 Implement the use of "this"</p> <p>4.6 Implement the use of abstract classes</p> <p>4.7 Implement the use of threads</p> <p>4.8 Demonstrate the Activity lifecycle to implement a constantly running game loop using the run method.</p> <p>4.9 Move objects around the screen</p> <p>4.10 Implement collision detection techniques based on overlapping rectangles</p>
<p>5. Understand the concept of Java Arrays</p>	<p>5.1 Implement a Java array</p> <p>5.2 Implement the enhanced for each loop to process the content of an array holding multiple instances of the same</p> <p>5.3 Implement sophisticated bitmap graphics</p> <p>5.4 Insert timings into a game</p>
<p>6. Understand the concept of Java ArrayLists</p>	<p>6.1 Implement the matrix class to reverse the direction of a bitmap image</p> <p>6.2 Implement the enumeration class</p>
<p>7. Create sophisticated interactive games using Design Patterns and object-oriented techniques to build a hierarchy of classes to avoid code duplication and large classes</p>	<p>7.1 Implement design patterns</p> <p>7.2 Implement interfaces to communicate effectively between classes</p> <p>7.3 Implement interfaces to communicate effectively between classes</p> <p>7.4 Implement a general-purpose sound engine</p> <p>7.5 Devise a method that will store the high score in</p>

		memory to preserve it when the game is replayed or the device is switched off
	7.6	Build a HUD (Heads Up Display) with control buttons
	7.7	Implement the Observer design pattern to enable the game engine to inform other classes when they need to handle user input
	7.8	Implement methods to replay, pause and resume game
8. Manage multi-touch inputs and create a particle system explosion effect	8.1	Implement the Observer pattern to create a multi-touch controller to listen for broadcasts from the game engine
	8.2	Write code to handle simultaneous touches of two or more control buttons
	8.3	Create multiple particles of varying size in an ArrayList
	8.4	Create a physics engine which will emit the particles in the ArrayList in random directions and speeds to create an explosion effect
9. Understand and use the Entity-Component and Factory design patterns to build objects from other objects	9.1	Create a generic game object that can be used as the basis for building objects with different characteristics from object specifications and component lists
	9.2	Implement the concept of composition over inheritance
	9.3	Create an object “factory” to build a variety of different game objects from the corresponding specifications and components
10. Build a sophisticated Space Wars game with a scrolling background, where the player controls a spaceship which dodges a variety of alien ships and attempts to shoot them down	10.1	Implement the matrix method to create a reverse image of a suitable background image for use in a scrolling background
	10.2	Artificially join the background image and its mirror image to enable the joint image to be moved either to the left or to the right, depending on which way the player’s ship is facing
	10.3	Implement several different alien objects, with different movement capabilities and speeds, to be built in the game object factory
	10.4	Manage multiple collision scenarios between lasers and ships and ships hitting ships, including the implementation of explosion effects
	10.5	Implement multiple game levels based on the player’s performance