Government of **Western Australia**
School Curriculum and Standards Authority

SAMPLE TEACHING AND LEARNING OUTLINE

TECHNOLOGIES

DIGITAL TECHNOLOGIES

YEAR 5

2020/10737v4

This document is an introduction to planning a teaching and learning outline with syllabus content for Year 5 Digital Technologies. It provides suggested sequencing and timing for teaching the syllabus content. For further details on curriculum requirements and available options, teachers should refer to the School Curriculum and Standards Authority's (the Authority's):

- *Policy Standards for Pre-primary to Year 10: Teaching, Assessing and Reporting*
- *Table* 1: *Western Australian Curriculum and Assessment Outline*: curriculum requirements and available options.

Schools may choose to teach the syllabus content for two hours per week for a semester, **or** one hour per week for the year. Sample plans provide a range of possible learning experiences from which assessment should be drawn. This *Year 5 Sample Teaching and Learning Outline* provides teachers with possible learning experiences over eight weeks and unpacks the syllabus content to assist teachers in their understanding.

A presentation (*Western Australian Curriculum Technologies Presentation*), which unpacks the process to develop this plan, is available on the Presentations page of the Authority website (https://k10outline.scsa.wa.edu.au/home/resources/presentations).

Year 5 Syllabus Content – Digital Technologies

Content	Description
Digital systems	Digital systems have components with basic functions that may connect together to form networks which transmit data
Representation of data	Data is represented using codes
Collecting, managing and analysing data	Collect, store and present different types of data for a specific purpose using software
Digital implementation	Design solutions to a user interface for a digital system Design, follow and represent diagrammatically, a simple sequence of steps (algorithm), involving branching (decisions) and iteration (repetition) Implement and use simple programming environments that include branching (decisions) and iteration (repetition) Create and communicate information, including online collaborative projects, using agreed social, ethical and technical protocols (codes of conduct)
Investigating and defining	Define a problem, and set of sequenced steps, with users making a decision to create a solution for a given task Identify available resources
Designing	Develop and communicate alternative solutions and follow design ideas, using annotated diagrams, storyboards and appropriate technical terms
Producing and implementing	Select, and apply safe, procedures when using components and equipment to make solutions
Evaluating	Develop negotiated criteria to evaluate and justify design processes and solutions
Collaborating and managing	Work independently, or collaboratively when required, to plan, develop and communicate ideas and information for solutions

Year Level Description

In Year 5, students further develop understanding and skills in computational thinking, such as identifying similarities in different problems and describing smaller components of complex systems. They have opportunities to create a range of solutions, such as games and interactive stories and animations that involve branching (choice of options).

Students explore the role that individual components of digital systems play in the processing and representation of data. They learn to acquire, justify and track various types of data. Students are introduced to the concept of data states in digital systems and how data are transferred between systems.

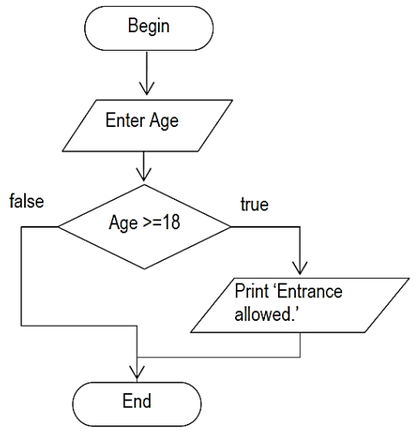
Students use abstractions by identifying common elements across similar problems and systems. They develop an understanding of the relationship between models and the real-world systems they represent.

When creating solutions, students identify appropriate data and requirements. They develop skills to write clear algorithms by identifying repetition and incorporate repeat instructions or structures when implementing their solutions. They make judgments about design solutions against the effectiveness in existing information systems.

Students develop strategies to communicate information and ideas using agreed ethical protocols, taking into account the safety aspects of working in digital environments.

Year 5 Learning Area: Technologies – Digital Technologies			
Year 5 Achievement Standard At Standard, students identify components of digital systems and their basic functions that connect to form networks which transmit data. They represent data using code, as well as using software to collect, store and present data for a specific purpose. Students create design solutions for a user interface and design, follow and represent diagrammatically, a simple sequence of steps (algorithms), involving branching (decisions) and iteration (repetition), implementing and using simple programming. They create and communicate information for online collaborative projects, using agreed social, ethical and technical protocols (codes of conduct). In digital technologies, students define a problem, identify available resources and create algorithms (sequenced steps) to assist in decision making for a given digital task. They develop and communicate alternative solutions, and use annotated diagrams, storyboards and appropriate technical terms when following design ideas. Students select and apply safe procedures when using components and equipment. They develop negotiated criteria to evaluate and justify design processes and solutions. Students work independently, or collaboratively, to plan, safely develop and communicate ideas and information.			
Weeks	**Syllabus content**	**Content unpacked**	**Suggested teaching and learning experiences**
1–2	**Digital systems** Digital systems have components with basic functions that may connect together to form networks which transmit data	• All systems involve an input, a process and an output. • Distinguish between internal components and external components, such as: ▪ internal: central processing unit, power supply, video card ▪ external: mouse, monitor, keyboard, printer, scanner. • Digital systems involve inputting data through a range of peripherals, processing through computer hardware, such as Central Processing Units and primary storage, and outputting information or data through output devices. • Peripherals are nonessential to the running of a device or computer system, but adds useability for the end user. • Input devices include keyboards, mice, barcode scanners, game controllers, scanners etc. • Output devices include monitors, printers, speakers etc. • These devices can be wired or wireless. Data needs to be transmitted to these devices to receive data. • A network refers to more than one device connecting and communicating together. • Peer-to-peer networks – no central server with devices having the same privileges and access as each other. • Client-server networks – a server attached to the network and other devices can share the server's resources. The server sets the restrictions and protocols that are to be used on the network.	• Create a list of hardware and compare to a list of peripherals (categorisation game). • Explain the functions of peripherals. • Explains the different types of networks. • Identify what devices are wired and what devices communicate wirelessly. • Open a physical computer case and label the components. Explain the interconnectedness of the how the components transmit to each other. Useful links: • Input and output game (https://www.abcya.com/games/input_output). • Connecting Physical Components (https://www.digitaltechnologieshub.edu.au/teachers/scope-and-sequence/5-6/digital-systems/connecting-digital-components).
3–4	**Representation of data** Data is represented using codes **Designing** Develop and communicate alternative solutions and follow design ideas, using annotated diagrams, storyboards and appropriate technical terms	• Computers can only process and send data in whole numbers. These are referred to as binary and are represented as ones (1) or zeros (0). • These are on or off electronic pulses. • All data that is represented will be processed to binary regardless of being text, sound, video, image etc. • Diagrammatic designs include storyboards, flowcharts, site maps etc. • All diagrams need written text to inform the purpose or explain details. These can include labels or annotations. • When technical terms are used, an explanation should be given.	• Brainstorm codes that students may be familiar with; for example, Morse code, barcodes and QR codes. • Explain that binary is a method used to convert data into a form that a computer will recognise and be able to process. • Converting numbers and text to and from binary/ASCII code using given tables. • Students use the ASCII Encoder Cards from the previous letter to write a message in code and then share with others to decode the message. • Students complete a series of multiple-choice questions (or a worksheet) on digital systems, components and data representation. Can be completed online. Useful links: • Use binary code to create an on/off picture (http://www.digitaltechnologieshub.edu.au/teachers/lesson-ideas/using-binary-to-create-on-off-pictures). • Complete Lesson 1 Binary Baubles (https://code.org/files/CSEDbinary.) – convert text to binary using ASCII Encoder cards. • Pixels and binary digits (https://www.digitaltechnologieshub.edu.au/teachers/assessment/assessment-ideas/pixels-and-binary-digits).

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences
5	**Collecting, managing and analysing data** Collect, store and present different types of data for a specific purpose using software	• Data gathering can be from primary or secondary data sources. Primary data is collected by the students themselves. Secondary data is collected from third party sources and is used to analyse a situation or problem. • Interpretation of data refers to the reviewing of collected data to view patterns and trends and, with this knowledge, make assumptions about the data or future data. • The manipulation of data can be the changing of data based on parameters given or retrieving the data that is required.	• Students select a topic of interest from the Australian Bureau of Statistics data packs (https://www.abs.gov.au/websitedbs/D3310114.nsf/Home/2016%20DataPacks). • Students then evaluate with the aid of graphing data to present interesting facts to the class via an oral presentation.
6–8	**Digital implementation** Design solutions to a user interface for a digital system Design, follow and represent diagrammatically, a simple sequence of steps (algorithm), involving branching (decisions) and iteration (repetition) Implement and use simple programming environments that include branching (decisions) and iteration (repetition) **Investigating and defining** Define a problem, and set of sequenced steps, with users making a decision to create a solution for a given task Identify available resources	• Sequence of steps refers to the order of a program. This must be logical to avoid any logic errors. • Branching is a term that refers to having more than one intended outcome. • The digital solution must have choices. These are called selection. The three types of selection are: ▪ one-way selection (if-then) ▪ two-way selection (if-then-else) ▪ multiway selection (CASE). • Iteration: ▪ test first (do while) ▪ test last (repeat Loop) ▪ fixed (for loop). • Simple visual programming environments include simple block-based programming, such as Scratch™. This allows students to learn the skills of computational thinking without the constraints of syntax rules, such as spelling. • Inputs can be received in a variety of ways, including mouse, keyboard, speech and touchscreen. • Technical protocols refer to the set of rules agreed to by students when using online collaboration. These can simply include respecting hardware, IT code of conduct or netiquette. • Management of this information should include the validity and referencing of any data gathered. • Common symbols in a flowchart and what they represent, for example: ▪ start and end (rounded rectangle or oval) ▪ process (rectangle) ▪ decision (diamond) ▪ input/output (parallelogram) ▪ line connectors (arrow). • Common content included in a storyboard, such as: ▪ sketches of key scenes/slides ▪ main event in each scene or content of slides ▪ actions that will take place ▪ audio and/or text to be added ▪ sequence is the logical order – computers work sequentially, line by line, top to bottom. If the order is incorrect a logic error will be given. Students cannot put shoes on before socks, a computer works in the same way. Students need to understand that logical sequence is very important in programming. • Solutions need to be innovative and incorporate problem-solving skills. • Available resources should include hardware and software. Students need to justify why they have chosen the selected hardware or software.	• Individual activity – Journaling toward algorithmic thinking (https://docs.google.com/document/d/1Qouj-ZxcPVmYehvIvLGnNV0X_4E_9YNyjXEeCOmmBaI/edit#heading=h.trqxxjck5x44). • Lego™ (or block) model activity [may be completed using actual blocks or Lego Digital Design software (https://www.lego.com/en-us/ldd/download)]. ▪ Students write their own series of instructions to create a simple Lego (or block) model. ▪ Swap instructions with another student. ▪ Without asking questions, students are to create the model from the given instructions. ▪ Compare the model created with the original model – did the instructions provide all the necessary detail to create the model accurately? Useful links: • Lucidchart – Sample Flowchart Template (https://www.lucidchart.com/pages/templates/flowchart/sample-flowchart-template) • Lucidchart – Computer Flowchart Template (https://www.lucidchart.com/pages/templates/flowchart/computer-flowchart-template) • BBC Northern Ireland – Sample Storyboard (http://www.bbc.co.uk/northernireland/myplacemyspace/promote-your-day-out/with-film/sample-storyboard.shtml). • Given the flowchart provided, discuss if students think that they would be able to follow the plan? • Cut out large shapes on cardboard and add content to the shapes when relevant. • Students create a flowchart and/or storyboard to demonstrate a process or design an animation Useful link: • Alice – Tutorial: Designing and Animation (https://www.alice.org/resources/exercise-and-project/tutorial-designing-an-animation/) .

<table>
<tr><th>Weeks</th><th>Syllabus content</th><th>Content unpacked</th><th>Suggested teaching and learning experiences</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Flowchart</th><th>Pseudocode</th></tr>
<tr><td> 
</td><td>One way selection
If condition then

Input (Age)
If Age >= 18 then
Output (‘Entrance allowed’)
End If</td></tr>
</table>
</td></tr>
<tr><td>9–11</td><td>Digital implementation
Implement and use simple programming environments that include branching (decisions) and iteration (repetition)</td><td>• A visual programming language is one that allows the user to create a program or code using graphic elements rather than text, for example:
▪ Scratch
▪ Kodu™
▪ Alice™
▪ Blocky Games: Turtle ™.
• Begin to introduce subject and software specific terminology as appropriate to selected software and task, for example:
▪ programming
▪ code
▪ loops/iteration
▪ branching/decisions.</td><td>• Introduce selected visual programming tools/software and explore the user interface.
• Practise simple algorithms to create a visual effect; for example, draw a simple shape or move an object around the screen.
• Students complete selected tutorials to develop their programming skills in the selected software, for example:
▪ Scratch (https://scratch.mit.edu/help/videos/)
▪ Kodu (https://www.kodugamelab.com/)
▪ Alice (http://www.alice.org/resources/alice-3-how-tos/)
▪ Blocky Games Turtle (https://blockly-games.appspot.com/turtle?lang=en&level=1).

Useful link:
• Create a basic game (https://www.digitaltechnologieshub.edu.au/teachers/scope-and-sequence/5-6/creating-digital-solutions/creating-a-digital-game).</td></tr>
<tr><td>11–14</td><td>Producing and implementing
Select, and apply safe, procedures when using components and equipment to make solutions</td><td>• When producing solutions students are required to do so safely. Safe use of hardware and software should represent the schools ICT code of conduct and the classroom rules set by the teacher.
• A storyboard is a graphical plan of an intended product, such as a video or animation.</td><td>• Students create a short film to show the correct and incorrect way of implementing the schools ICT code of conduct.
▪ Storyboards should be created before filming.
▪ Filming to be completed in small groups.
▪ Editing can be completed using a variety of software, including:
○ iMovie™
○ Blender™
○ Lightworks™
○ Quik (App based, iOS and Android)™.</td></tr>
</table>

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences
		Storyboard example  	

15–16	**Evaluating**
	Develop negotiated criteria to evaluate and justify design processes and solutions
	Collaborating and managing
	Work independently, or collaboratively when required, to plan, develop and communicate ideas
		• Evaluation can be formative, summative or both.
		• Developing criteria for evaluation should be conducted as a class or individually.
		• Peer evaluation of designs is very beneficial to students.
			• Film festival – students watch and evaluate all the short films previously created.
			  ▪ Students decide on assessment criteria for their films.
			  ▪ Students evaluate each other's films.
			• Students then write 'what would I do differently' responding to feedback given.

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences
	and information for solutions		
17–18	**Digital implementation** Create and communicate information, including online collaborative projects, using agreed social, ethical and technical protocols (codes of conduct)	• Discuss copyright status of online images, including: ▪ usage rights of images ▪ suitable free copyright sites for images • breaches of copyright law.	• Teacher demonstration of selecting images at different copyright rights from google images and explains each of the options. • Activity: 'Let's vote on it!' (https://k10outline.scsa.wa.edu.au/home/assessment/assessment-activities/year5). Useful link: • Digital Citizenship™ lesson idea (https://www.digitaltechnologieshub.edu.au/teachers/scope-and-sequence/5-6/collaboration-and-protocols/digital-citizenship).

Note: the above Teaching and Learning program is based on two hours per week for 18 weeks for a total of 36 hours.