

# Curriculum planning for Digital Technologies 7–10

Teaching and assessing in the  
Digital Technologies Curriculum

# Copyright

© Victorian Curriculum and Assessment Authority (VCAA) 2021.

VCAA presentations may be reproduced in accordance with the [VCAA's Copyright Policy](#), and as permitted under the *Copyright Act 1968*.

# In this session

## Background to teaching the Digital Technologies Curriculum

- Planning a lesson
- Scope and sequence 7–10
- Content descriptions
- Elaborations
- Unpacking the content descriptions
- Plugged and unplugged activities
- Unit plan ideas
- Lesson planning

## Background to assessing the Digital Technologies Curriculum

- Achievement standards
- Assessing student work
- Connections between the content descriptions and the achievement standards

# Background to teaching the Digital Technologies Curriculum 7–10

# Planning a lesson

- Start with the Digital Technologies Curriculum.
- Determine what needs to be taught using the content descriptions.
- Develop some ideas for activities by referring to the:
  - Elaborations
  - Unpacking the Content Descriptions resource
  - Unit plan ideas.
- Plan for assessing student learning using the achievement standards.

# Scope and sequence 7-10

	Foundation – Level 2	Levels 3 and 4	Levels 5 and 6	Levels 7 and 8	Levels 9 and 10
<b>Digital Systems</b>	Identify and explore digital systems (hardware and software components) for a purpose	Explore a range of digital systems with peripheral devices for different purposes, and transmit different types of data	Examine the main components of common digital systems, and how such digital systems may connect together to form networks to transmit data	Investigate how data are transmitted and secured in wired, wireless and mobile networks	Investigate the role of hardware and software in managing, controlling and securing the movement of and access to data in networked digital systems
<b>Data and Information</b>	Recognise and explore patterns in data and represent data as pictures, symbols and diagrams Collect, explore and sort data, and use digital systems to present the data creatively Independently and with others create and organise ideas and information using information systems, and share these with known people in safe online environments	Recognise different types of data and explore how the same data can be represented in different ways Collect, access and present different types of data using simple software to create information and solve problems Individually and with others, plan, create and communicate ideas and information safely, applying agreed ethical and social protocols	Examine how whole numbers are used as the basis for representing all types of data in digital systems Acquire, store and validate different types of data and use a range of software to interpret and visualise data to create information Plan, create and communicate ideas, information and online collaborative projects, applying agreed ethical, social and technical protocols	Investigate how digital systems represent text, image and sound data in binary Acquire data from a range of sources and evaluate their authenticity, accuracy and timeliness Analyse and visualise data using a range of software to create information, and use structured data to model objects or events Manage, create and communicate interactive ideas, information and projects collaboratively online, taking safety and social contexts into account	Analyse simple compression of data and how content data are separated from presentation Develop techniques for acquiring, storing and validating quantitative and qualitative data from a range of sources, considering privacy and security requirements Analyse and visualise data to create information and address complex problems, and model processes, entities and their relationships using structured data Manage and collaboratively create interactive solutions for sharing ideas and information online, taking into account social contexts and legal responsibilities
<b>Creating Digital Solutions</b>	Follow, describe and represent a sequence of steps and decisions (algorithms) needed to solve simple problems	Define simple problems, and describe and follow a sequence of steps and decisions involving branching and user input (algorithms) needed to solve them Design a user interface for a digital system, generating and considering alternative design ideas Design, modify and follow simple algorithms represented diagrammatically and in English, involving sequences of steps, branching, and iteration	Define problems in terms of data and functional requirements, drawing on previously solved problems to identify similarities Design the user experience of a digital system, generating and communicating alternative designs Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors Develop and modify programs with user interfaces involving branching, iteration and functions using a programming language	Define and decompose real-world problems taking into account functional requirements and sustainability (economic, environmental, social), technical and usability constraints Design the user experience of a digital system, evaluating alternative designs against criteria including functionality, accessibility, usability and aesthetics Design algorithms represented diagrammatically and in structured English and validate algorithms and programs through tracing and test cases Develop modular programs, applying selected algorithms and data structures including using an object-oriented programming language	Define and decompose real-world problems precisely, taking into account functional and non-functional requirements and including interviewing stakeholders to identify needs Design the user experience of a digital system, evaluating alternative designs against criteria including functionality, accessibility, usability and aesthetics Design algorithms represented diagrammatically and in structured English and validate algorithms and programs through tracing and test cases Develop modular programs, applying selected algorithms and data structures including using an object-oriented programming language Evaluate critically how well student-developed solutions and existing information systems meet needs, are innovative and take account of future risks and sustainability
<b>Achievement Standard</b>	By the end of Level 2, students identify how common digital systems are used to meet specific purposes. Students use digital systems to represent simple patterns in data in different ways and collect familiar data and display them to convey meaning. Students design solutions to simple problems using a sequence of steps and decisions. They create and organise ideas and information using information systems and share these in safe online environments.	By the end of Level 4, students describe how a range of digital systems and their peripheral devices can be used for different purposes. Students explain how the same data sets can be represented in different ways. They collect and manipulate different data when creating information and digital solutions. They plan and safely use information systems when creating and communicating ideas and information, applying agreed protocols. Students define simple problems, and design and develop digital solutions using algorithms that involve decision-making and user input. They explain how their developed solutions and existing information systems meet their purposes.	By the end of Level 6, students explain the functions of digital systems components and how digital systems are connected to form networks that transmit data. Students explain how digital systems use whole numbers as a basis for representing a variety of data types. They manage the creation and communication of ideas, information and digital projects collaboratively using validated data and agreed protocols. Students define problems in terms of data and functional requirements and design solutions by developing algorithms to address the problems. They incorporate decision-making, repetition and user interface design into their designs and explain their digital solutions, including a iterations, and develop, test, and modify digital solutions. Their developed solutions meet current and future needs taking sustainability into account.	By the end of Level 8, students distinguish between different types of networks and their suitability in meeting defined purposes. Students explain how text, image and sound data can be represented and secured in digital systems and presented using digital systems. They analyse and evaluate data from a range of sources to model solutions and create information. They manage the collaborative creation of interactive ideas, information and projects and use appropriate codes of conduct when communicating online. Students define and decompose problems in terms of functional requirements and constraints. They design user experiences and algorithms incorporating branching and iterations, and develop, test, and modify digital solutions. Students evaluate information systems and their solutions in terms of meeting needs, innovation and sustainability.	By the end of Level 10, students explain the control and management of networked digital systems and the data security implications of the interaction between hardware, software and users. Students explain simple data compression, and why content data are separated from presentation. They take account of privacy and security requirements when selecting and validating data and use digital systems to analyse, visualise and model salient aspects of data. Students show and conduct when communicating online protocols for the legal and safe use, transmission and maintenance of data and projects. Students define and decompose complex problems in terms of functional and non-functional requirements. They design and evaluate user experiences and algorithms, and develop and test modular programs, including an object-oriented program. Students evaluate their solutions and information systems in terms of risk, sustainability and potential for innovation.

Content description

Achievement standard

# Content descriptions

Show more

## Levels 7 and 8 Content Descriptions

---

### Digital Systems

---

Investigate how data is transmitted and secured in wired, wireless and mobile networks (VCDTDS035)

- Content descriptions are what is required to be taught.
- They are divided into bands.
- Content descriptions can be taught individually or as part of a wider unit with multiple content descriptions.

# Elaborations

## Digital Technologies / Levels 7 and 8 / Digital Systems

Content description	Elaborations
Investigate how data is transmitted and secured in wired, wireless and mobile networks	<ul style="list-style-type: none"><li>explaining that networks have components that control the movement of data, for example routers, hubs, switches and bridges manage data traffic and that the characteristics of these components impact on the operation (speed and security) of networks</li><li>explaining how cellular radio towers (transceivers) and mobile phones work together to create mobile networks</li><li>comparing the reliability and speed of transmitting data through wireless, wired and mobile networks</li><li>recognising that there are different communications protocols for transmitting data in networks, for example hypertext transfer protocol (HTTP) is used for transferring web page files in a browser, file transfer protocol (FTP) is used for sending and receiving any files over a network and transmission control protocol/internet protocol (TCP/IP) is used for controlling file transfers over the internet</li></ul>
<b>Code</b> VCDTDS035	
<b>Curriculum resources and support</b>	

- Each content description links to elaborations.
- Elaborations can be used a guide for activities to teach content.
- They are not meant to be prescriptive.



# Unpacking the content descriptions

## Digital Technologies: Unpacking the Content Descriptions

Strand	Creating Digital Solutions	Sample activities
<b>Content Description</b>	Define and decompose real-world problems taking into account functional requirements and sustainability (economic, environmental, social), technical and usability constraints	<ul style="list-style-type: none"><li>• decomposing a problem into the smallest component parts, and classifying parts, such as a website's menus, pages and buttons, or a game's characters, objects and backgrounds</li><li>• exploring what a program or solution must do to function as intended, for example comparing a user's input to a stored username and password combination, or passing data from one function to another</li><li>• investigating materials that are used to create digital systems and the environmental impact of their creation</li><li>• following the path to recycling or disposal and the costs associated with each</li><li>• define the intended audience for the program or solution, the assumptions that have been made about age, physical dexterity, vision and hearing, then detail where adjustments could be incorporated on the proposed device or platform</li></ul>
<b>Related extract from Achievement Standard</b>	Students define and decompose problems in terms of functional requirements and constraints.	
<b>Suggested focus</b>	Lessons may focus on: <ul style="list-style-type: none"><li>• decomposition and abstraction</li><li>• functional requirements</li><li>• sustainability issues in technology, such as:<ul style="list-style-type: none"><li>• energy consumption</li><li>• materials production</li><li>• end of use options</li></ul></li><li>• assumptions about physical capability of users</li></ul>	

# Plugged vs unplugged activities

## Plugged activities

- Plugged activities are activities that require students to use digital systems, including hardware and software or the internet.

## Unplugged activities

- Unplugged activities are activities that do not require students to use digital systems or the internet.

# Unplugged activities

Levels 7 and 8

## Creating Digital Solutions

---

Design the user experience of a digital system, generating, evaluating and communicating alternative designs

---

Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors

## Suggested activities

---

- Design the user interface for a new computer game, using a storyboard to explain the stages of the game.
- Identify the features of the new game that will make it effective, such as the storyline, gameplay, environment, goals and rewards.
- Write an algorithm for a task, such as entering in a class set of test scores. Write the algorithm as English statements and include decisions and repetition.
- Test the algorithm by tracing the input of a few test scores to check for the accuracy of the algorithm (desk-checking). Identify any errors and modify the algorithm.

# Plugged activities

Levels 9 and 10

## Digital Systems

---



Investigate the role of hardware and software in managing, controlling and securing the movement of and access to data in networked digital systems

## Suggested activities

---

- Use software such as Visio to create a network diagram of a building in the school or the school network. Clearly identify network components (such as servers and routers, devices and transmission media).
- Conduct research online to compare the similarities and differences between two operating systems, such as Windows and macOS. Use software such as Word, Pages or Google Docs to identify and describe a range of characteristics of the operating system, including processing, control, storage and security.

# Unit plan ideas

 VICTORIAN CURRICULUM AND ASSESSMENT AUTHORITY 

## Digital Technologies, Foundation to Level 10 – Unit plan ideas

The VCAA has developed the following resource for Digital Technologies. The resource includes unit plans that cover one strand and its associated content descriptions from Foundation to Level 10. The unit plans include ideas for learning activities and assessment activities that could be completed by students at home, if they are working remotely, or by students working in the classroom.

This resource has been developed to assist teachers with ideas for planning and delivering a series of lessons that meet the content descriptions and then assessing student work against the relevant achievement standard. The unit plans are grouped by band: F–2, 3–4, 5–6, 7–8 and 9–10. Each unit plan covers one strand: Digital Systems, Data and Information, or Creating Digital Solutions. Relevant extracts from the achievement standard have been provided.

Each of the unit plans includes a sample timeline and ideas for lesson activities and assessment activities. Each unit plan also contains a link to the relevant VCAA curriculum area plan resource, for teachers who want to develop units of work based on these resources.

The ideas in these unit plans involve students identifying and exploring hardware, software and network components; transmission media and network security; drawing diagrams; listing items and writing descriptions; recording their voice; conducting surveys; using spreadsheets; collaborating online; using computational and design thinking involving problem solving to determine requirements; interviewing stakeholders; generating designs and evaluation criteria; designing algorithms; developing programs; and evaluating solutions.

Teachers would need to prepare students before delivering a unit of work based on these unit plans. They could use a range of teaching and learning activities (see the [Plugged activities and Unplugged activities](#) on the Digital Technologies [Curriculum advice for remote and flexible learning page](#)). They may also wish to prepare student worksheets to go with the ideas in the unit plans.

Teachers should monitor students' completion of a unit of work and assess against the relevant achievement standard. Students could submit evidence for teachers to assess by taking photographs of their work and emailing them to the teacher; emailing completed documents or solutions to the teacher; or uploading images and documents to the school learning management system.

© VCAA

Digital Technologies, Foundation to Level 10 – Unit plan ideas

### Digital Technologies, Levels 9 and 10 – Unit plan ideas

**Achievement standard extracts**

**By the end of Level 10, ...**

Students define and decompose complex problems in terms of functional and non-functional requirements. They design and evaluate user experiences and algorithms, and develop and test modular programs, including an object-oriented program. Students evaluate their solutions and information systems in terms of risk, sustainability and potential for innovation.

**Strand and content descriptions**

**Creating Digital Solutions**

Define and decompose real-world problems precisely, taking into account functional and non-functional requirements and including interviewing stakeholders to identify needs [VCOTC0056](#)

Design the user experience of a digital system, evaluating alternative designs against criteria including functionality, accessibility, usability and aesthetics [VCOTC0057](#)

Design algorithms represented diagrammatically and in structured English and validate algorithms and programs through tracing and test cases [VCOTC0058](#)

Develop modular programs, applying selected algorithms and data structures including using an object-oriented programming language [VCOTC0059](#)

Evaluate critically how well student-developed solutions and existing information systems and policies take account of future risks and sustainability and provide opportunities for innovation [VCOTC0054](#)

**Ideas for delivery of unit**

The unit may have 3 × 45-minute lessons each week over nine weeks.

<b>Week 1: Decompose problems</b>	<ul style="list-style-type: none"><li>Introduce students to breaking down complex problems into smaller problems.</li><li>Define functional and non-functional requirements in developing a solution.</li><li>Students interview stakeholders to identify their needs and the functional and non-functional requirements of the solution.</li><li>Assessment activity (refer to the assessment activities below).</li></ul>
<b>Week 2: Design the user experience</b>	<ul style="list-style-type: none"><li>Create two or more designs of user interfaces for an app or a game.</li><li>Develop evaluation criteria for selecting the best design.</li><li>Consider a wide range of users, including people with disabilities.</li><li>Assessment activity (refer to the assessment activities below).</li></ul>
<b>Week 3: Design algorithms</b>	<ul style="list-style-type: none"><li>Create simple algorithms and revise control structures.</li><li>Design algorithms that use functions and data structures.</li><li>Define the rules of Structural English (pseudocode).</li><li>Develop algorithms that incorporate basic object-oriented programming concepts, such as calling a method and using object properties.</li><li>Test the expected output of algorithms using tracing and desk checking, making modifications and recording results.</li><li>Assessment activity (refer to the assessment activities below).</li></ul>

© VCAA Page 9

Unit plan ideas could be used as:

- a sequence of lessons
- a complete unit of work
- assessment ideas.

# Unit plan ideas

Digital Technologies, Levels 9 and 10 – Unit plan ideas

## Achievement standard extracts

By the end of Level 10, ...

Students define and decompose complex problems in terms of functional and non-functional requirements. They design and evaluate user experiences and algorithms, and develop and test modular programs, including an object-oriented program. Students evaluate their solutions and information systems in terms of risk, sustainability and potential for innovation.

# Unit plan ideas

## Strand and content descriptions

### Creating Digital Solutions

Define and decompose real-world problems precisely, taking into account functional and non-functional requirements and including interviewing stakeholders to identify needs ([VCDTCD050](#))

Design the user experience of a digital system, evaluating alternative designs against criteria including functionality, accessibility, usability and aesthetics ([VCDTCD051](#))

Design algorithms represented diagrammatically and in structured English and validate algorithms and programs through tracing and test cases ([VCDTCD052](#))

Develop modular programs, applying selected algorithms and data structures including using an object-oriented programming language ([VCDTCD053](#))

# Unit plan ideas

## Ideas for delivery of unit

### Creating Digital Solutions

The unit may have 3 × 45-minute lessons each week over nine weeks.

#### Week 1: Decompose problems

- Introduce students to breaking down complex problems into smaller problems.
- Define functional and non-functional requirements in developing a solution.
- Students interview stakeholders to identify their needs and the functional and non-functional requirements of the solution.
- Assessment activity (refer to the assessment activities below)



# Unit plan ideas

## Ideas for delivery of unit

### Week 2: Design the user experience

- Create two or more designs of user interfaces for an app or a game.
- Develop evaluation criteria for selecting the best design.
- Consider a wide range of users, including people with disabilities.
- Assessment activity (refer to the assessment activities below)

### Week 3: Design algorithms

- Create simple algorithms and revise control structures.
- Design algorithms that use functions and data structures.
- Define the rules of Structured English (pseudocode).
- Develop algorithms that incorporate basic object-oriented programming concepts, such as calling a method and using object properties.
- Test the expected output of algorithms using tracing and desk checking, making modifications and recording results.

# Unit plan ideas

## Ideas for delivery of unit

### Weeks 4–8: Develop modular programs

- Review a general-purpose programming language by creating functions with arguments that return a value.
- Identify and record the objects, events and properties in favourite games or apps.
- Describe how methods are used in a simple program.
- Create more complex programs that use methods and object properties.
- Create modules that use classes, methods and object properties.
- Progressively test the functionality of the program using a testing table to check actual output versus expected output and make modifications to the module where appropriate.
- Assessment activity (refer to the assessment activities below)

# Unit plan ideas

## Ideas for delivery of unit

### Weeks 9: Evaluate student- developed solutions

- Discuss how the student-developed solution meets the functional and non-functional requirements of the stakeholder.
- Investigate sustainability issues with digital solutions, such as e-waste, compatibility, energy use and redundancy.
- Compare student-developed solutions with existing products and discuss potential room for innovation.
- Develop evaluation criteria as a class to assess student-developed solutions that take into account user experience, original requirements and accessibility.
- Assessment activity (refer to the assessment activities below)

# Unit plan ideas

## Ideas for assessment activities

### Pre-test

- A simple written test on terminology and concepts from the Levels 7 and 8 Creating Digital Solutions strand

### Decompose problems

- A written test on terminology and concepts
- A written report on a game or application proposal that includes functional and non-functional requirements
- A visual diagram showing the top-down design of a complex problem
- A transcript of an interview with a stakeholder discussing their needs and identifying their functional and non-functional requirements

# Unit plan ideas

## Ideas for assessment activities

### Design the user experience

- A written report of a specification of a software solution to be developed
- A table with a list of evaluation criteria used to evaluate the completed software solution
- Two or more mock-ups of the proposed user interfaces
- A written report evaluating the mock-ups and a justification of the chosen mock-up to be created as a software solution

### Design algorithms

- A written test on rules of flowcharts and Structured English
- A workbook with algorithm solutions to given tasks in class
- A portfolio of student-developed algorithms, including tracing and modifications

# Unit plan ideas

## Ideas for assessment activities

### Develop modular programs

- A written test on terminology and concepts
- A portfolio of working software solutions and testing tables for each modular program
- An annotated visual report of a student's analysis of an existing game or application
- A student proposal for a developed game or application

### Evaluate student-developed solutions

- A written test on sustainability issues with digital technologies
- A written report comparing student-developed solutions with existing solutions
- A written evaluation report containing an evaluation table and conclusion that assesses a student's developed solution against the functional and non-functional requirements and the stakeholder's needs

# Lesson planning

Term 3	
Week	
1	Algorithms: Conventions
2	Algorithms: Control structures
3	Algorithms: Testing and tracing
4	Algorithms: Solving problems (Assessment)
5	Programming: Languages and environment
6	Programming: Interpreting algorithms
7	Programming: Data structures

## Week 2 Control structures

### Learning intention

Explain how control structures are used in algorithms and programming.

### Success criteria

I can identify sequence, iteration and branching structures.  
I can design algorithms that use different controls structures.

### Warm up

Write down the steps required to wash dishes.

...

# Lesson planning

## Week 2 Control structures

Warm up

Write down the steps required to wash dishes.

...

Introduction

Videos and Images of control structures.

Student activities

.....

## Week 2 Control structures

Student activities:

- Provide examples of algorithms.
- Students correctly identify control structures.
- Provide worded and visual problems.
- Students design algorithms.
- Review – Provide a review question to clarify learning.



# Background to assessing the Digital Technologies Curriculum 7–10

# Achievement standards

## Levels 7 and 8

By the end of Level 8, students distinguish between different types of networks and their suitability in meeting defined purposes.

Students explain how text, image and sound data can be represented and secured in digital systems and presented using digital systems. They analyse and evaluate data from a range of sources to model solutions and create information. They manage the collaborative creation of interactive ideas, information and projects and use appropriate codes of conduct when communicating online.

Students define and decompose problems in terms of functional requirements and constraints. They design user experiences and algorithms incorporating branching and iterations, and develop, test, and modify digital solutions. Students evaluate information systems and their solutions in terms of meeting needs, innovation and sustainability.

## Levels 9 and 10

By the end of Level 10, students explain the control and management of networked digital systems and the data security implications of the interaction between hardware, software and users.

Students explain simple data compression, and why content data are separated from presentation. They take account of privacy and security requirements when selecting and validating data and use digital systems to analyse, visualise and model salient aspects of data. Students share and collaborate online, establishing protocols for the legal and safe use, transmission and maintenance of data and projects.

Students define and decompose complex problems in terms of functional and non-functional requirements. They design and evaluate user experiences and algorithms, and develop and test modular programs, including an object-oriented program. Students evaluate their solutions and information systems in terms of risk, sustainability and potential for innovation.

# Making connections between content descriptions and achievement standards



# About the achievement standards

- They represent the student's level of achievement.
- They are what the student should be able to demonstrate.
- They are an indication of where a student should be at on the continuum of learning.
  
- They are not the same as the content descriptions.

# Assessing student work

- Relate the assessment tasks to the content taught.
- Account for the different levels of achievement.
- Consider the achievement standards above and below the target level.
- Write in a language that students can understand.
- Share the assessment rubric or marking scheme with students.

# Review

## Background to teaching the Digital Technologies Curriculum

- Planning a lesson
- Scope and sequence 7–10
- Content descriptions
- Elaborations
- Unpacking the content descriptions
- Plugged and unplugged activities
- Unit plan ideas
- Lesson planning

## Background to assessing the Digital Technologies Curriculum

- Achievement standards
- Assessing student work
- Connections between the content descriptions and the achievement standards

# Thank you

## Contact

Phil Feain

Digital Technologies Curriculum Manager, VCAA

[Philip.Feain@education.vic.gov.au](mailto:Philip.Feain@education.vic.gov.au)

Ph: (03) 9059 5146

