Digital Technologies – 7 _ Data and Information

	Strand		Knowledge and understanding				Processes and production skills													
							Creating digital solutions by:													
			Digital systems		Representation of data		Collecting, managing and analysing data		Investigating and defining				Generating and designing				Producing and implementing		Evaluating		Collaborating and managing	
	Content Description		Investigate how data is transmitted and secured in wired, wireless and mobile networks, and how the specifications affect performance (ACTDIK023)		Investigate how digital systems represent text, image and audio data in binary (ACTDIK024)		Acquire data from a range of sources and evaluate authenticity, accuracy and timeliness (ACTDIP025)		Analyse and visualise data using a range of software to create information, and use structured data to model objects or events (ACTDIP026)		Define and decompose real-world problems taking into account functional requirements and economic, environmental, social, technical and usability constraints (ACTDIP027)		Design the user experience of a digital system, generating, evaluating and communicating alternative designs (ACTDIP028)		Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors (ACTDIP029)		Implement and modify programs with user interfaces involving branching, iteration and functions in a general-purpose programming language (ACTDIP030)		Evaluate how student solutions and existing information systems meet needs, are innovative, and take account of future risks and sustainability (ACTDIP031)		Plan and manage projects that create and communicate ideas and information collaboratively online, taking safety and social contexts into account (ACTDIP032)	
Sequence of Lessons / Unit	**Approx. time rq'd**	**Year A or B**	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #
	10	7	☐		☐		☑	7	☑	7	☑	4	☑	7	☐		☐		☑		☑	8

Years 5 and 6 Achievement Standard	**Years 7 and 8 Achievement Standard**	**Years 9 and 10 Achievement Standard**
By the end of Year 6: • Students explain the fundamentals of digital system components (hardware, software and networks) and how digital systems are connected to form networks. (1) • They explain how digital systems use whole numbers as a basis for representing a variety of data types. (2) • Students define problems in terms of data and functional requirements and design solutions by developing algorithms to address the problems. (3) • They incorporate decision-making, repetition and user interface design into their designs and implement their digital solutions, including a visual program. (4) • They explain how information systems and their solutions meet needs and consider sustainability. (5) • Students manage the creation and communication of ideas and information in collaborative digital projects using validated data and agreed protocols. (6)	By the end of Year 8 1. Students distinguish between different types of networks and defined purposes. 2. They explain how text, image and audio data can be represented, secured and presented in digital systems. 3. Students plan and manage digital projects to create interactive information. 4. They define and decompose problems in terms of functional requirements and constraints. 5. Students design user experiences and algorithms incorporating branching and iterations, and test, modify and implement digital solutions. 6. They evaluate information systems and their solutions in terms of meeting needs, innovation and sustainability. 7. They analyse and evaluate data from a range of sources to model and create solutions. 8. They use appropriate protocols when communicating and collaborating online.	By the end of Year 10 • Students explain the control and management of networked digital systems and the security implications of the interaction between hardware, software and users. (1) • They explain simple data compression, and why content data are separated from presentation. (2) • Students plan and manage digital projects using an iterative approach. (3) • They define and decompose complex problems in terms of functional and non-functional requirements. (4) • Students design and evaluate user experiences and algorithms. (5) • They design and implement modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities. (6) • They take account of privacy and security requirements when selecting and validating data. Students test and predict results and implement digital solutions. (7) • They evaluate information systems and their solutions in terms of risk, sustainability and potential for innovation and enterprise. (8) • They share and collaborate online, establishing protocols for the use, transmission and maintenance of data and projects. (9)

Data and information
This sequence uses the context of meal planning to demonstrate a process to solve a problem; in this case, what meal to cook for teenagers with various needs. In decomposing the problem, students collaborate to better understand their audience's needs, the food options available and to define how the functional requirements of the solution can be met. Key to meeting the functional requirements is collecting data related to suggested meals and organising this as structured data so it can be collected, sorted and visualised in different ways. Students also need to consider one or more of the following constraints when designing their solution: sustainability (economic, environmental, social), technical considerations and usability. Teachers may substitute their own context and follow a similar process described in this sequence.

Flow of activities				
Short text	**Decompose the problem** Break down the problem into key parts and consider the functional requirements.	**Collect and input data** Acquire data from different sources and put it into an online spreadsheet or database to share information and create a large database of information.	**Designing an app** Design an app that solves the problem.	**Evaluate the process and the design** Evaluate the design as well as the process used to gather data and collaborate online.
Questions to guide exploration	*Why and how do you decompose a problem?*	*How can you store and share data in an online spreadsheet?*	*How do I design a graphical user interface?*	*Was the design successful and does it meet the audience needs? How well did the group perform in completing the project?*
AC Alignment	*Investigating and defining (ACTDIP027)*	*Investigating and defining (ACTDIP026)* *Collecting, managing and analysing data (ACTDIP025)*	*Generating and designing (ACTDIP028)* *Evaluating (ACTDIP031)*	*Evaluating (ACTDIP031)*
What's this about?	Defining and decomposing problems are part of the analysis process. Once you have stated the problem, usually in terms of a brief description of the problem's elements and the stakeholders involved, you can begin to break the problem down into smaller elements. This helps reduce the complexity of problems because you can see their sub-tasks or sub-elements. It allows you to get a better understanding or insight into the problem and therefore its solution. It is like asking a set of smaller questions so that the larger question (or problem) can be answered. You are really identifying a set of needs, which in turn might help decide what data is needed to solve the problem. Breaking a problem down into smaller parts allows you to study each part in more detail and reduces the problem's complexity. Decomposition also allows you to identify connections between elements; some will be strong and others not, as well as showing the importance of any constraints on the solution. Tools such as decision trees and fishbone diagrams can be used to decompose a problem.	An important aspect of data entry in a spreadsheet is the process of validating data. One way to validate the data is to create and use a drop down menu to limit free text entry. Examples might include: • **cost**: the range might include low, medium and high, where: – low = < $10 – med = $10–$20 – high = > $20 • **health rating**: a star rating out of 5 or colour (red, orange, green). Spreadsheets enable us to record, sort and analyse data and also to visualise that data in different ways to make sense of patterns or trends. A database is similar in many ways to a spreadsheet in that it stores data in structured ways, and the data can be queried and reported. Usually databases store more data than spreadsheets and the data is often connected to other data (relational). Large databases enable us to	Paper prototyping is ideal for conceptualising a design; for example, an interface for an app. It is a quick way to document potential designs and to consider the user experience. It is also a useful way to consider a list of requirements and how they are to be met, and to address the identified constraints.	**Design evaluation** The primary focus of design evaluation is to assess whether the design meets the needs of the target audience. Does the interface enable the user to easily locate and select information that is of interest? **Student performance evaluation** Typically, teachers are required to report on students' performance. Digital technologies work is often project-based and a variety of pieces of evidence are needed to assess students' performance.

		locate information on the internet, as search strings act as queries on the data.		
The focus of the learning (in simple terms)	Use a suitable hook to engage students in a relevant context and problem. For example, students could practise decomposing a problem by finding a holiday that would suit a friend or family member. Students break the problem down to identify the needs of the solution. They can do that by asking the following questions: • What is the age group of the person? • Are there any preferred modes of travel (eg plane, train, car, boat)? • Are there any price restrictions? • What are the preferred activities (eg adventure, sightseeing) • What is the preferred style of accommodation? • What is the best time for the person to travel? • How many days can the person be on holidays? • Are there any special needs, such as wheelchair access? Based on this introductory activity, students can transfer their skills to decomposing the following problem: What meals are suitable for teenagers with various needs? It is suggested that students be provided with at least one constraint on the solution; for example: sustainability (economic, environmental, social), technical considerations or usability. The following are some questions relevant to this problem. • What types of meals appeal to teenagers? • What are the challenges (eg dietary requirements, cooking skills)? • What are the preferred processes (eg frying, grilling, steaming)? • Are there preferred cuisines (eg Mexican, Thai, Indian)? • Are there nutritional requirements? Do they differ for males and females? • What are the favourite ingredients? • What is the favourite meal? Breakfast, brunch, lunch, dinner? • How much can be spent on ingredients for a meal? • How can information be accessed easily anytime, anywhere to help with decisions about suitable meals? Organise students into groups to decompose the problem. Once students have decomposed the problem, they need to know what data is required in a solution so that a teenager could find a meal that met their needs.	Students collect recipe and menu data from relevant websites, cookbooks or from family. They estimate the cost based on the ingredients. In groups, students create a spreadsheet that records relevant data so that a teenager can find a meal that suits their particular needs. Ideally, students would combine their menu data to efficiently build a large database of information. As a method of collating suggested menu items and accompanying data use a collaborative tool such as Google Sheets. Model how to structure data with relevant ways to validate the data. This makes data entry easier and limits errors in the data. For each heading, agree as a class on the range for each and show how to create the menu items for each column. Discuss how to enter free text such as ingredients and menu items to limit errors. Test the spreadsheet using a couple of examples and adjust if necessary. Visualise the data; for example, create a chart to analyse coverage across the various categories to determine if there are instances of no options. In this case, extra suitable menu items may be added. You could discuss limitations of the spreadsheet and discuss how a database might be a better option to store the data. You could also convert the data in the spreadsheet to an online shareable database such as the add-on Airtable or Obvibase. Alternatively, the data could be added directly to an online or computer-based database software package. Students could test out search queries using their database; for example: 1. What menu items have a heath rating above 4, are Asian cuisine, vegetarian, medium cost and easy to cook? 2. What menu items are gluten free?	Students design an app interface. They are not creating the app, although that could be an option if they have the skills, interest and time. Discuss common databases used every day through a graphical user interface (GUI) such as Netflix, Spotify or eBay. Discuss which elements of user interface design make them easy to use. Provide examples for students to view and evaluate or, if access is limited, model a suitable application using a data projector. Students refer to the meal planner data and how this might be incorporated into the design of an app. In groups they collaborate to generate several design options that cater for the target audience. They refer back to the constraint and the other needs of the problem when determining their preferred design idea. Ask students to create a paper prototype. To support this process, provide print-outs of a tablet or smartphone screen for students to draw their designs on. Ask students to annotate what each screen enables the user to do. This might be undertaken as an individual task first and then each student could share their design with the group to create a final design. Students think about how the user would search and locate relevant options and see relevant information when selecting from options. In their app they could consider: • graphical elements • home screen • details page for each menu item • a rating system • a form of artificial intelligence (AI) that would offer suggestions based on user behaviour • sharing information such as an ingredients list via email/SMS.	**Design evaluation** How well does the final design meet the needs of the intended audience? Students ask someone to use the paper prototype and to provide feedback on the design. What questions arise from the user as they progress? Do they get confused? At what point? **Student performance evaluation** What evidence can students provide that demonstrates their contribution to the project? When reflecting on the project what have they learned?

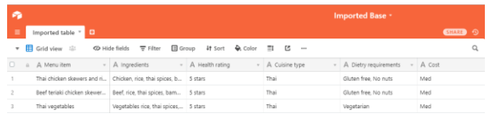
<table>
<tr>
<td></td>
<td>
Build a picture of the needs of teenagers by considering ‘types’ within the group ‘teenagers’. Types may include those who are health conscious, are athletes, have allergies and special dietary requirements, have a preferred cuisine, etc.

In defining the problem further, students consider options such as:
• meal cost
• time and effort to prepare and cook the meal
• maximum number of ingredients
• image of the plated meal.

Generate a class list of ‘types and options’ that can be used to gather data for the design. Model using two to three meal suggestions to define a structured data table. For example:
<table>
<tr><th>Menu item</th><th>Ingredients</th><th>Health rating</th><th>Cuisine type</th><th>Dietary requirements</th><th>Cost</th><th>Skill level</th><th>Cook time</th></tr>
<tr><td>Thai chicken skewers and rice</td><td>Chicken, rice, Thai spices, bamboo skewers</td><td>5 stars</td><td>Thai</td><td>Gluten free
Nut free</td><td>Med</td><td>Easy</td><td>15–20 min</td></tr>
<tr><td>Beef teriyaki and rice</td><td>Beef, rice, teriyaki sauce</td><td>5 stars</td><td>Japanese</td><td></td><td>Med</td><td>Easy</td><td>15–20 min</td></tr>
<tr><td>Thai vegetables</td><td>Vegetables, Thai spices</td><td>5 stars</td><td>Thai</td><td>Gluten free
Nut free
Vegetarian</td><td>Med</td><td>Easy</td><td>15–20 min</td></tr>
</table>
Another possible context is to provide questions around access to digitised music. A relevant data set is provided.

Optional
You could ask students to evaluate existing apps to examine app interfaces and rate the usefulness of design and access to information. Students could document three key design-related learnings that could influence their design of a digital solution.

This task could also be undertaken in conjunction with a food technology/home economics class where students research a suitable menu item. Alternatively, as a task completed at home, students could select and prepare a meal to taste-test a chosen menu item. The task could also have a health focus and students could create a healthy breakfast calculator using spreadsheeting software. As an example, refer to the Healthy breakfast calculator spreadsheet.
</td>
<td>
3. What menu items are low cost and Indian, Thai or Japanese cuisine?



Screen shot of Airtable.
</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supporting resources and tools and purpose/ context for use</td>
<td>
The 10 best apps every foodie must have on their phone
Download some of these apps to evaluate and investigate user interface and how data is displayed and presented.

Million song dataset
This is a freely available collection of audio features and metadata for a million contemporary popular music tracks.
</td>
<td>
Google Sheets
Create a shareable online spreadsheet.

Create a drop down menu in Excel
This webpage demonstrates how to create a drop down menu with easy to follow screenshots and an explanation.
</td>
<td>
Paper prototyping: The 10-minute practical guide
Learn about paper prototyping.

POP
POP helps you to transform pen and paper ideas into an interactive iPhone or Android prototype.
Mockdrop
</td>
<td>
Evaluation and assessment: Part 1
This is an overview of assessment ideas and methods for computational thinking.

Evaluation and assessment: Part 2
This second video investigates how rubrics can be used to support understanding of the problem-
</td>
</tr>
</table>

	There can only be one This lesson sequence takes students through defining a problem, acquiring and analysing data and implementing and evaluating a programming solution. Decision tree software Use this software to create your own decision tree diagram. Cut through the technical language of digital technologies Use this resource as an example of decomposing a problem and using a decision tree.	Airtable A free trial version of this online database tool is available. Schools would need to purchase it. This tool works in Google Drive, provides support and guidance, and is suitable for students to learn about databases. Obvibase This online database tool works in Google Drive. It provides limited support and guidance and relies on students knowing how to create a database from a blank sheet. Introduction to databases (SQL) This brief tutorial teaches the basics of using Structured Query Language (SQL) – a special purpose programming language that is used to communicate with databases.	This is a free website to create mock-ups for different devices.	solving process. It discusses how students could engage with decomposition and abstraction.
Assessment	**Suggested approaches** • List of key steps when decomposing the problem • Level of engagement when discussing audience needs **Achievement standard** **Define** and **decompose** problems in terms of functional requirements and constraints. **Analyse** and **evaluate** data from a range of sources to **model** and **create** solutions.	**Suggested approaches** • Contribution to online spreadsheet or database • Discuss search queries in relation to database **Achievement standard** **Analyse** and **evaluate** data from a range of sources to **model** and **create** solutions.	**Suggested approaches** Students explain their individual design drawings compared with the final design. **Achievement standard** **Design** user experiences and algorithms incorporating branching and iterations, and **test**, **modify** and **implement** digital solutions.	**Suggested approaches** • Evaluation of the final design and how well it meets the needs of the intended audience • Reflection of the project and what the student has learned **Achievement standard** **Evaluate** information systems and their solutions in terms of meeting needs, innovation and sustainability.