Digital Technologies

2018 Subject Outline

Stage 1

This Board-accredited Stage 1 subject outline

will be taught from **2018.**

Published by the SACE Board of South Australia,  
60 Greenhill Road, Wayville, South Australia 5034

Copyright © SACE Board of South Australia 2017

First published 2017

Published online November 2017  
ISBN 978 1 74102 839 3 (online Microsoft Word version)

ref: A671676

*This subject outline is accredited for teaching at Stage 1 from 2018*

contents

Introduction 1

Subject description 1

Capabilities 2

Aboriginal and Torres Strait Islander knowledge, cultures, and perspectives 5

Learning scope and requirements 6

Learning requirements 6

Content 6

Assessment scope and requirements 18

Evidence of learning 18

Assessment design criteria 18

School assessment 19

Performance standards 21

Assessment integrity 24

Support materials 25

Subject-specific advice 25

Advice on ethical study and research 25

Introduction

Subject description

Digital Technologies is a 10-credit subject or a 20-credit subject at Stage 1, and a 20‑credit subject at Stage 2.

Digital technologies have changed the ways that people think, work, and live. The application of digital technologies can lead to discoveries, new learning, and innovative approaches to understanding and solving problems.

The study of Digital Technologies provides a platform for deep interdisciplinary learning. Students make connections with innovation in other fields and across other learning areas.

In Digital Technologies students create practical, innovative solutions to problems of interest. By extracting, interpreting, and modelling real-world data sets, students identify trends and examine sustainable solutions to problems in, for example, business, industry, the environment, and the community. They investigate how potential solutions are influenced by current and projected social, economic, environmental, scientific, and ethical considerations, including relevance, originality, appropriateness, and sustainability.

Innovation in Digital Technologies involves students creating new ways of doing things, generating their own ideas and creating digital solutions to problems of interest. Solutions may take the form of a product, prototype, and/or proof of concept. Students are encouraged to experiment and learn from what does not work as planned, as well as from what does work. Innovation may also include students designing solutions that improve existing processes or products.

Students use computational thinking skills and strategies to identify, deconstruct, and solve problems that are of interest to them. They analyse and evaluate data, test hypotheses, make decisions based on evidence, and create solutions. Through the study of Digital Technologies, students are encouraged to take ownership of problems and design, code, validate, and evaluate their solutions. In doing so, they develop and extend their understanding of designing and programming, including the basic constructs involved in coding, array processing, and modularisation.

At Stage 1, students develop and apply their skills in computational thinking and in program design. They follow agile practices and/or iterative engineering design processes. Learning environments in Digital Technologies may include physical, online, and/or simulated spaces.

Digital Technologies promotes learning through initiative, collaboration, creativity, and communication using project- and inquiry-based approaches.

Capabilities

The capabilities connect student learning within and across subjects in a range of contexts.

The SACE identifies seven capabilities. They are:

* literacy
* numeracy
* information and communication technology (ICT) capability
* critical and creative thinking
* personal and social capability
* ethical understanding
* intercultural understanding.

Literacy

In this subject students extend and apply their literacy capability by, for example:

* using digital and other communication strategies to engage in collaborative projects and tasks
* interpreting, analysing, and evaluating primary and secondary data sets relevant to problems of interest
* using a range of visualisation tools to identify and deconstruct problems and design solutions (product, prototype, and/or proof of concept), incorporating the terminology and conventions of digital technologies
* becoming competent and confident designers of the features of digital solutions
* using accurate and appropriate terminology to interpret, discuss, and explain concepts, issues, problems, and solutions involving digital technologies
* communicating project ideas using tools such as storyboards and other visual representations
* reading and interpreting online documentation and tutorial materials that support coding.

Numeracy

In this subject students extend and apply their numeracy capability by, for example:

* using computational thinking in identifying, deconstructing, and solving problems of interest
* predicting trends by analysing data sets
* displaying numerical and statistical data in appropriate formats, using visualisation tools
* examining the usefulness of results and preparing validation plans for calculating outputs of digital solutions
* using code that enables manipulation of numerical data in digital solutions
* applying appropriate mathematical and logical concepts and thinking in programming.

Information and communication technology (ICT) capability

In this subject students extend and apply their ICT capability by, for example:

* locating and accessing information using digital technologies
* using a range of digital technologies to extract, interpret, and model data sets
* presenting findings using multimodal approaches, including infographics
* using computational thinking skills and strategies to identify, deconstruct, and solve problems of interest
* using program-design skills to develop digital solutions (product, prototype, and/or proof of concept)
* using digital technologies to work collaboratively on various projects, including creating innovative solutions
* understanding the impact of ICT, its development and application, and how it shapes and is shaped by social agendas.

Critical and creative thinking

In this subject students extend and apply their critical and creative thinking capability by, for example:

* visualising possibilities and scoping solutions
* using computational thinking skills to analyse and evaluate data, and make predictions and decisions
* identifying and deconstructing problems of interest
* using abstraction to deconstruct problems
* exploring and testing hypotheses using design-related investigations
* using initiative in designing projects and solutions
* critically evaluating existing and proposed digital solutions
* thinking creatively about ways of working collaboratively to design digital solutions
* designing innovative, creative, and appropriate digital solutions using agile practices
* applying programming as a creative tool.

Personal and social capability

In this subject students extend and apply their personal and social capability by, for example:

* working collaboratively to identify team roles, protocols, and deliverables
* collaborating with stakeholders to create and evaluate innovative digital solutions
* making decisions and taking initiative in designing projects
* exploring innovation and developing entrepreneurial skills
* sharing and discussing ideas about problems, progress, and innovative solutions
* listening to and respecting the perspectives of others
* acquiring practical skills, knowledge, and understanding related to the design, development, and use of digital solutions
* forming a critical understanding of how digital solutions affect individuals, groups, and/or societies
* planning effectively and managing time.

Ethical understanding

In this subject students extend and apply their ethical understanding capability by, for example:

* considering the ethical implications of digital solutions on individuals, groups, and/or societies now and in the future
* considering ethical strategies for working with individuals and groups
* evaluating the ethical issues surrounding the collection, storage, and use of data
* making ethical decisions based on evidence
* using data and reporting the outcomes accurately and fairly
* recognising the importance of responsible participation in social, economic, environmental, scientific, and/or ethical decision-making
* increasing critical understanding of the appropriate and ethical uses of digital technology
* understanding the implications of data use for personal, social, and ethical participation in the wider community
* evaluating the reliability of information for accurate decision-making.

Intercultural understanding

In this subject students extend and apply their intercultural understanding capability by, for example:

* exploring design issues in local, national, and global contexts to expand knowledge of, and create solutions for, a diverse range of individuals, groups, and societies
* respecting and engaging with different cultural views and customs, and exploring these interactions using digital technologies
* recognising that engaging with different perspectives enhances own knowledge, understanding, and solutions
* being open-minded and receptive to changes in design and project development
* understanding that the progress of designing and implementing a digital solution influences and is influenced by cultural factors.

Aboriginal and Torres Strait Islander knowledge, cultures, and perspectives

In partnership with Aboriginal and Torres Strait Islander communities, and schools and school sectors, the SACE Board of South Australia supports the development of high-quality learning and assessment design that respects the diverse knowledge, cultures, and perspectives of Indigenous Australians.

The SACE Board encourages teachers to include Aboriginal and Torres Strait Islander knowledge and perspectives in the design, delivery, and assessment of teaching and learning programs by:

* providing opportunities in SACE subjects for students to learn about Aboriginal and Torres Strait Islander histories, cultures, and contemporary experiences
* recognising and respecting the significant contribution of Aboriginal and Torres Strait Islander peoples to Australian society
* drawing students’ attention to the value of Aboriginal and Torres Strait Islander knowledge and perspectives from the past and the present
* promoting the use of culturally appropriate protocols when engaging with and learning from Aboriginal and Torres Strait Islander peoples and communities.

Learning scope and requirements

Learning requirements

The learning requirements summarise the knowledge, skills, and understanding that students are expected to develop and demonstrate through their learning in Stage 1 Digital Technologies.

In this subject, students are expected to:

1. apply computational thinking skills to explore problems and possible solutions

2. develop and apply programming skills in creating digital solutions

3. analyse patterns and relationships in data sets and/or algorithms, and draw conclusions

4. develop and apply program-design skills to create and evaluate digital solutions

5. research and discuss ethical considerations in digital technologies

6. work individually and collaboratively.

Content

Stage 1 Digital Technologies may be undertaken as a 10-credit subject or a 20-credit subject. The subject consists of the following focus areas:

* Focus area 1: Programming
* Focus area 2: Advanced programming
* Focus area 3: Data analytics
* Focus area 4: Exploring innovations.

For a 10-credit subject, students study at least two focus areas. For a 20-credit subject, students study at least three focus areas.

The focus areas are not intended to be taught independently or to be of equivalent length. They should be sequenced and structured to suit individual cohorts of students.

Computational thinking skills are integral to each focus area, together with applying program-design skills and exploring innovation. Students analyse patterns and relationships in data sets and/or algorithms, and draw conclusions about their usefulness in defining the problem.

In developing and applying their program-design skills, students develop and extend their understanding of program-design methodology. They take a structured approach to designing an algorithm or digital solution (product, prototype, and/or proof of concept) that is appropriate to the context of the problem and meets the needs of the intended user. They code, test, and evaluate their solutions.

In creating and/or evaluating their solutions, students take into account ethical considerations. These may include, for example, implications of data use and/or digital solutions for individuals, groups, societies, and/or the environment.

(Note: a digital solution may be implemented by the student or may be a working prototype or proof of concept.)

In Focus area 1: Programming, students identify and deconstruct a problem, and develop and use code to design and test possible solutions. In Focus area 2: Advanced programming, students extend their programming skills with a particular focus on problem-solving. In Focus area 3: Data analytics, students apply their computational thinking skills to analyse relationships in data sets, identify and scope problems, and create solutions. In Focus area 4: Exploring innovations, students apply their critical and creative thinking skills to explore digital innovations, develop ideas, and create digital solutions.

Pivotal to student learning is the development of the capabilities and, in particular, the skills for creativity, collaboration, and innovation.

Students develop and apply their critical and creative thinking in Digital Technologies through visualising possibilities, exploring innovations, and creating digital solutions. They engage with innovators and centres of innovation as they explore how to be creative and innovative in their own work. Collaboratively and individually students generate, develop, and refine their ideas, exploring and investigating possible solutions before developing a product, prototype, or proof of concept. Students make connections in their learning and use their initiative to design and create innovative digital solutions.

Students work collaboratively to problem-solve and create digital solutions to problems of interest. Collaboration enables students to develop their social and cognitive skills and ways of combining the knowledge and skills of group members to identify and solve problems effectively as a team. Students build their personal and social capability as they work together in interconnected ways, with their activities reliant on contributions from others and building on activities begun by others. Through collaboration students extend and apply their critical and creative thinking, decision-making, and problem-solving skills and capabilities. Collaboration may include the use of synchronous and/or asynchronous digital communication strategies.

Innovation in Digital Technologies includes students generating and articulating their own ideas, creating new processes, products, and solutions, and designing solutions that improve existing ideas, processes, and products. Students develop and extend the range and combination of skills that enable them to contribute to innovation in digital solutions. These skills encompass academic, technical, and soft skills, and the ability to apply these skills and knowledge to solving unfamiliar problems. Students work independently and/or collaboratively to generate ideas and create innovative solutions and creative products. They extend and apply their skills in critical and creative thinking and problem-solving, and make connections in their learning across disciplines to generate ideas and create innovative digital solutions. In the pursuit of innovation students increase their willingness to take risks and appreciate the value of learning from what does not work, as well as from what does work, as they scope and design innovative solutions.

Focus area 1: Programming

Students use computational thinking skills and strategies to understand problems and explore possible solutions. These strategies include deconstructing problems into logical sets of sequential or iterative parts, using tools that show dependencies and flow of sequence between subparts.

Students design algorithms to produce an output. An algorithm might be written in plain English or pseudocode, or be represented in a flow chart. Students turn the algorithm into code and test their solutions, which may take the form of a product, prototype, and/or proof of concept.

Students learn, develop, and practise with the building blocks of a programming language (variables, expressions, assignment, and input/output commands), understand the concept of sequencing, and use the constructs of selection and iteration.

They test and evaluate their solutions using simple testing techniques, and consider how effectively their solution meets the needs of the intended user.

Self-assessment tools or skills frameworks may be used to support the development and application of students’ skills in working collaboratively.

The following framework provides a set of possible techniques and strategies that can be used for learning.

| Key learning elaboration | Possible techniques and strategies |
| --- | --- |
| **Computational thinking** |  |
| Students apply computational thinking skills to design solutions prior to coding. | Students use decomposition to deconstruct a problem logically into two or more sequential tasks using tools, including structure charts, flow charts, and/or mind maps. |
| Students work collaboratively and individually to design a solution. | Students apply logical thinking to deduce as much new information as possible from the initial information provided, and revisit steps based on new information.  Students use pattern recognition to simplify algorithmic solutions through iteration. |
| **Design thinking** |  |
| Students use computational thinking skills to identify and define problems. | Students identify and define problems, questions, or hypotheses. |
| Students design and implement algorithms. | Students outline solutions to develop algorithms.  Students test algorithms.  Students code algorithms. |
| **Programming** |  |
| Students learn simple programming skills. | Students learn and use variables, expressions, and assignment.  Students learn and use data types (strings, numbers, and Booleans).  Students learn and use input/output commands, understand the concept of sequencing, and use the constructs of selection and iteration.  Students develop and extend understanding of the building blocks of a general-purpose programming language (GPL).  Students use software development tools such as integrated development environments (IDEs) and compilers/interpreters to support code development. |
| **Testing and evaluating** |  |
| Students test and evaluate their solution. | Students learn and use simple tests to detect logic and runtime errors such as tracing variables and running test cases.  Students evaluate their solution with system testing.  Students consider how effectively the solution meets the needs of the intended user. |

Focus area 2: Advanced programming

This focus area builds on the knowledge, skills, and understanding developed in Focus area 1: Programming or previous learning.

Students extend their computational thinking skills and strategies to understand a range of problems, and explore and code possible solutions. These strategies include abstract and diagrammatic reasoning (to deconstruct a problem and focus on core concepts and ideas including simulation), data manipulation (moving from simple to more complex structures), and basic optimisation.

Students investigate problems that are of interest to them, and develop their solutions iteratively. They identify patterns in algorithms, and develop, test, and code their own algorithms. A solution may take the form of a product, prototype, and/or proof of concept.

In Focus area 2 students extend their programming knowledge and coding skills. This may include moving from a simple visual programming language to a text programming language; or extending the practice of the first programming language with more sophisticated programming, such as modularisation, code reuse, and the use of data structures such as arrays.

Students identify and select strategies to solve problems and generate programming solutions. This provides opportunities for students to work collaboratively and to apply project design skills and innovation. Students test and evaluate their programs, and research and discuss ethical considerations.

Self-assessment tools or skills frameworks may be used to support the development and application of students’ skills in working collaboratively.

The following framework provides a set of possible techniques and strategies that can be used for learning.

| Key learning elaboration | Possible techniques and strategies |
| --- | --- |
| **Problem-solving** |  |
| Students learn and apply problem-solving strategies. | Students consider problem-solving strategies such as mathematical equations, simulation, and greedy or brute-force algorithms.  Students interpret problem descriptions and identify the connection between input data and desired outputs. |
| Students develop solutions for a range of numerical and text manipulation problems. | Students use diagrams or logic to reason about the problem, identify constraints, and refine solutions.  Students work collaboratively.  Students reflect on the problem-solving process.  Students use abstraction to generalise solutions. |
| **Design thinking** |  |
| Students use computational thinking skills to identify and define problems. | Students identify, investigate, and define problems, questions, or hypotheses. |
| Students develop solutions. | Students develop solutions iteratively and incrementally, identifying scope for innovation.  Students outline solutions and develop algorithms. |
| Students design and implement complex algorithms. | Students test algorithms.  Students code algorithms.  Students present the solution, including documentation such as appropriate use of comments.  Students present a simulation of the solution. |
| **Programming** |  |
| Students code a range of algorithms using a general-purpose programming language (GPL). | Students code small programs and functions to implement well-defined tasks, such as finding if a number is a prime or if two words are anagrams. |
| Students read, store, and manipulate input data from text files. | Students accept simple formatted files as inputs and store the input in appropriate data structures such as arrays or lists.  Students write small programs to confidently use the syntax and semantics of basic commands. |
| Students master conditional statements, iteration, and vector operations through repeated practice. | Students improve the code through optimisation. |
| **Testing and evaluating** |  |
| Students learn to methodically test their programs. | Students practise program testing with a range of provided values.  Students write own test values and identify value range and border cases. |
| Students use a range of techniques to debug their code. | Students use print statements, paper tracing, module testing, and debugger tools to identify bugs and correct code.  Students test each function or code section separately. |
| **Ethical considerations** |  |
| Students research ethical implications. | Students research and discuss the ethical considerations in digital solutions for individuals, groups, societies, and/or the environment. |

Focus area 3: Data analytics

Students use computational thinking skills and strategies to analyse relationships in data sets, apply programming and program-design skills, and use a digital system to transform data into information.

They develop skills in identifying patterns of similarities or repetition in data sets, and making predictions and drawing conclusions from these patterns. They use simple techniques to analyse and display the data.

Students apply their computational thinking skills to identify and define problems using data, and develop solutions. They apply their programming and program-design skills to create and refine digital solutions. A solution may take the form of a product, prototype, and/or proof of concept.

Students have opportunities to work collaboratively and reflect on ways in which their solution could be used or improved.

Students extend their ethical understanding by researching and discussing the possible ethical implications of data collection, storage, and/or use.

Self-assessment tools or skills frameworks may be used to support the development and application of students’ skills in working collaboratively.

The following framework provides a set of possible techniques and strategies that can be used for learning.

| Key learning elaboration | Possible techniques and strategies |
| --- | --- |
| **Data analysis** |  |
| Students apply computational thinking to identify and analyse relationships in data sets. | Students discuss sources of data.  Students recognise patterns in data.  Students draw conclusions from patterns in data.  Students use simple techniques to analyse, tabulate, and display data.  Students investigate storage and management of data.  Students explore practical applications of data analytics. |
| **Design thinking** |  |
| Students use computational thinking skills to identify and define problems. | Students identify and define problems, questions, or hypotheses using data that can be collected or comes from publicly available data sets.  Students work collaboratively.  Students outline solutions to develop algorithms.  Students develop solutions iteratively and incrementally. |
| **Creating digital solutions** |  |
| Students create digital solutions based on data. | Students apply programming skills to create solutions such as dynamic webpages, application programs, and databases.  Students present solutions with appropriate documentation.  Students reflect on findings, and identify possible uses or innovation. |
| **Ethical considerations** |  |
| Students research and evaluate the ethical implications surrounding the collection, storage, use, and/or security of data. | Students research and evaluate strategies for use by government, business, organisations, and/or individuals for:   * collection of personal data * data security and storage * data protection and backup * privacy and anonymising data * cyber security. |

Focus area 4: Exploring innovations

This focus area is designed to link closely with each of the other focus areas. Students extend and apply their critical and creative thinking capability through exploring innovation, including developing an understanding of what makes a solution innovative and how to be innovative in their own work. They identify and deconstruct problems of interest, visualise possibilities, and think about ways of working creatively and collaboratively to scope and design digital solutions.

Students investigate current innovations and advances in technology. They are encouraged to work collaboratively and use their initiative and creativity to generate and develop ideas and turn these ideas into practical solutions.

Students integrate their learning from other focus areas by applying their computational thinking skills, program-design skills, programming skills, and/or data analysis skills to exploring innovative and creative solutions. A solution may take the form of a product, prototype, and/or proof of concept.

Students research ethical implications of digital solutions and possible ways in which solutions created now may be used in the future. They extend and apply their intercultural understanding by recognising how engaging with different perspectives enhances knowledge, understanding, and solutions.

Self-assessment tools or skills frameworks may be used to support the development and application of students’ skills in working collaboratively.

The following framework provides a set of possible techniques and strategies that can be used for learning.

| Key learning elaboration | Possible techniques and strategies |
| --- | --- |
| **Exploring innovations** |  |
| Students extend and apply critical and creative thinking skills. | Students explore and learn about critical and creative thinking. |
| Students develop an understanding of innovation. | Students explore the links between entrepreneurship and innovation.  Students engage with innovators and centres of innovation.  Students explore recent advances in technology.  Students consider how solutions that are created now might be used in the future; for example, embedded systems and the internet of things. |
| **Developing ideas** |  |
| Students generate ideas. | Students explore agile practices and how these facilitate innovative thinking and collaborative ways of working. |
| Students work collaboratively and communicate effectively. | Students use strategies to promote creativity and generate ideas.  Students use collaborative problem-solving skills to plan, structure, and develop an idea.  Students explore and analyse existing programs, applications, and digital solutions relevant to the idea.  Students refine the idea.  Students identify roles and activities for group members and map how these are interconnected.  Students agree on and apply digital and other communication strategies to enable participation and contributions from group members. |
| **Design thinking** |  |
| Students learn and apply project-design skills. | Students turn an idea into a solution such as a dynamic website, application program, wearable technology, or other digital solution.  Students define and follow an agreed project plan based on time available, communication strategies, and key features of deliverables. |
| **Creating a digital solution** |  |
| Students create a digital solution and reflect on its effectiveness. | Students create a digital solution or prototype.  Students reflect on the effectiveness of the solution and identify possible extensions or improvements. |
| **Ethical considerations** |  |
| Students research and discuss the ethical implications of digital solutions. | Students research and discuss ethical considerations such as intellectual property rights and copyright.  Students consider the ethical implications of digital solutions on individuals, groups, societies, and/or the environment. |

Assessment scope and requirements

Assessment at Stage 1 is school-based.

Evidence of learning

The following assessment types enable students to demonstrate their learning in Stage 1 Digital Technologies.

* Assessment Type 1: Project Skills
* Assessment Type 2: Digital Solution

For a 10-credit subject, students provide evidence of their learning through four assessments. Each assessment type should have a weighting of at least 20%. Students undertake:

* at least two project skills tasks
* at least one digital solution.

Students must have the opportunity to work collaboratively in at least one assessment.

For a 20-credit subject, students provide evidence of their learning through eight assessments. Each assessment type should have a weighting of at least 20%. Students undertake:

* at least four project skills tasks
* at least two digital solutions.

Students must have the opportunity to work collaboratively in at least two assessments.

Assessment design criteria

The assessment design criteria are based on the learning requirements and are used by teachers to:

* clarify for students what they need to learn
* design opportunities for students to provide evidence of their learning at the highest possible level of achievement.

The assessment design criteria consist of specific features that:

* students should demonstrate in their learning
* teachers look for as evidence that students have met the learning requirements.

For this subject the assessment design criteria are:

* computational thinking
* development and evaluation
* research and ethics.

The specific features of these criteria are described below.

The set of assessments, as a whole, must give students opportunities to demonstrate each of the specific features by the completion of study of the subject.

Computational Thinking

The specific features are as follows:

CT1 Application of computational thinking skills to explore problems and possible solutions.

CT2 Development and application of programming skills to create a digital solution or prototype.

CT3 Analysis of patterns and relationships in data sets and/or algorithms to draw conclusions.

Development and Evaluation

The specific features are as follows:

DE1 Development and application of program-design skills to create a digital solution or prototype.

DE2 Evaluation of the effectiveness of a digital solution or prototype.

DE3 Contribution to collaborative work.

Research and Ethics

The specific feature is as follows:

RE1 Research into and discussion of ethical considerations in digital solutions and/or data use.

School assessment

Assessment Type 1: Project Skills

For a 10-credit subject, students undertake at least two project skills tasks. For a 20‑credit subject, students undertake at least four project skills tasks.

Students examine approaches to identifying, deconstructing, and solving problems of interest by applying:

* computational thinking skills
* design and programming skills.

The tasks should focus on problems that the students identify as being of interest to them.

Tasks may be undertaken individually and/or collaboratively.

At least one of the tasks should involve students working collaboratively, with each student providing evidence of their role in and contribution to the collaborative task.

In at least one of the tasks, students should discuss ethical considerations, such as the implications of data use and/or digital solutions for individuals and/or groups.

Tasks should be presented in multimodal form, and each task should be the equivalent of a maximum of 5 minutes.

Tasks could include, for example:

* a screen-capture validation
* a multimedia presentation of the solution to a problem or analysis of data
* a prototype solution or proof of concept
* a debate about the ethics of data storage and use.

For this assessment type, students provide evidence of their learning primarily in relation to the following assessment design criteria:

* computational thinking
* development and evaluation
* research and ethics.

Assessment Type 2: Digital Solution

For a 10-credit subject, students undertake at least one digital solution. For a 20-credit subject, students undertake at least two digital solutions.

Students independently or collaboratively solve a problem. They use computational thinking skills and strategies to analyse relationships in data sets, apply programming and program-design skills, and use a digital system to transform data into information.

The solution may be a product, prototype, or proof of concept. Where a prototype is produced, there should be enough programming present to clearly explain how the solution will produce an output.

The problem should be chosen by, and be of interest to, the student(s).

The digital solution may include, for example:

* a website
* a program
* an application
* wearable technology.

Students evaluate the digital solution. The evaluation may include discussion about:

* a feature or features that could be considered innovative
* the effectiveness of the solution
* ethical considerations.

The evaluation may also include, depending on the focus areas studied:

* data analysis
* design and coding skills
* problem-solving skills.

The digital solution may be undertaken individually and/or collaboratively. If working collaboratively, each student presents an individual evaluation, including evidence of their contribution to the project and how they applied their collaborative skills.

The digital solution should be presented in digital or multimodal form, and the evaluation should be presented in multimodal, oral, or written form. The evaluation should be a maximum of 3 minutes if oral, 500 words if written, or the equivalent if multimodal.

For this assessment type, students provide evidence of their learning primarily in relation to the following assessment design criteria:

* computational thinking
* development and evaluation
* research and ethics.

Performance standards

The performance standards describe five levels of achievement, A to E.

Each level of achievement describes the knowledge, skills, and understanding that teachers refer to in deciding how well students have demonstrated their learning on the basis of the evidence provided.

During the teaching and learning program the teacher gives students feedback on their learning, with reference to the performance standards.

At the student’s completion of study of a subject, the teacher makes a decision about the quality of the student’s learning by:

* referring to the performance standards
* taking into account the weighting of each assessment type
* assigning a subject grade between A and E.

Performance Standards for Stage 1 Digital Technologies

| - | Computational Thinking | Development and Evaluation | Research and Ethics |
| --- | --- | --- | --- |
| A | Insightful and sustained application of computational thinking skills to explore problems and possible solutions.  Focused development and strategic application of a wide range of programming skills to create a digital solution or prototype.  In-depth analysis of patterns and relationships in data sets and/or algorithms to draw insightful conclusions. | Purposeful and well-considered development and application of program-design skills to create digital solutions or a prototype that include innovative features.  Insightful evaluation of the effectiveness of a digital solution or prototype.  Insightful and proactive contribution to collaborative work. | In-depth research into and discussion of the ethical considerations in digital solutions and/or data use. |
| B | Some insights in the application of computational thinking skills to explore problems and possible solutions.  Thorough development and well-considered application of a range of programming skills to create a digital solution or prototype.  Some depth in analysis of patterns and relationships in data sets and/or algorithms to draw well-informed conclusions. | Well-considered development and application of program-design skills to create digital solutions or a prototype that include one or more innovative features.  Well-considered evaluation of the effectiveness of a digital solution or prototype.  Mostly consistent and effective contribution to collaborative work. | Some depth in research into and discussion of the ethical considerations in digital solutions and/or data use. |
| C | Application of computational thinking skills to explore problems and possible solutions.  Competent development and application of programming skills to create a digital solution or prototype.  Description, with some analysis of patterns and relationships in data sets and/or algorithms, to draw generally informed conclusions. | Development and application of program-design skills to create digital solutions or a prototype that may include one or more innovative features.  Description, with some evaluation of the effectiveness, of a digital solution or prototype.  Effective contribution to collaborative work. | Considered research into and discussion of the ethical considerations in digital solutions and/or data use. |
| D | Some application of basic computational thinking skills to describe problems and possible solutions.  Basic development and some application of programming skills to create one or more partial solutions or prototypes.  Basic description of patterns and relationships in data sets and/or algorithms to draw one or more basic conclusions. | Some development and application of program-design skills to create one or more partial solutions or prototypes.  Basic description of a digital solution or prototype and one or more aspects of its effectiveness.  Some contribution to collaborative work. | Basic research into and discussion of the ethical considerations in digital solutions and/or data use. |
| E | Attempted application of a limited number of simple computational thinking skills to describe a problem and/or possible solution.  Attempted development and/or application of basic programming skills.  Attempted description of one or more patterns and relationships in data sets and/or algorithms. | Attempted development and application of program-design skills.  Attempted description of a digital solution or prototype.  Limited contribution to collaborative work. | Attempted discussion of an ethical consideration in digital solutions and/or data use. |

Assessment integrity

The SACE Assuring Assessment Integrity Policy outlines the principles and processes that teachers follow to assure the integrity of student assessments. This policy is available on the SACE website (www.sace.sa.edu.au) as part of the SACE Policy Framework.

The SACE Board uses a range of quality assurance processes so that the grades awarded for student achievement in the school assessment are applied consistently and fairly against the performance standards for a subject, and are comparable across all schools.

Information and guidelines on quality assurance in assessment at Stage 1 are available on the SACE website (www.sace.sa.edu.au).

Support materials

Subject-specific advice

Online support materials are provided for each subject and updated regularly on the SACE website (www.sace.sa.edu.au). Examples of support materials are sample learning and assessment plans, annotated assessment tasks, annotated student responses, and recommended resource materials.

Advice on ethical study and research

Advice for students and teachers on ethical study and research practices is available in the guidelines on the ethical conduct of research in the SACE, which are on the SACE website (www.sace.sa.edu.au).