

Dot Points for VCE Software Development¹

Software Development Unit 3

This unit focuses on the problem-solving methodology stages of analysis, design and development, with students learning how to interpret designs and develop solutions using a programming language; then analyse and design solutions to student-identified needs or opportunities.

Area of Study 1: Programming

- Learn the main concepts of programming (including GUI)
- School-assessed Coursework (SAC)
 - Develop a sequence of working modules from teacher provided designs

Key knowledge

Data and information

KK1: characteristics of data types

KK2: types of data structures, including one-dimensional arrays (single data type, integer index), associative arrays (or dictionaries or hash tables) and records (varying data types, field index)

Note: Stacks, Queues and Linked-Lists are mentioned under Data Structures in the Glossary

Approaches to problem solving

KK3: methods for documenting a problem, need or opportunity

KK4: methods for determining solution requirements, constraints and scope

KK5: methods of representing designs, including data dictionaries, object descriptions, mock-ups & pseudocode

KK6: formatting and structural characteristics of files, including delimited (CSV), plain text (TXT) and XML file formats

KK7: a programming language as a method for developing working modules that meet specific needs

KK8: naming conventions for solution elements

KK9: processing features of a programming language, including classes, control structures, functions, instructions and methods

KK10: algorithms for sorting, including selection sort and quicksort

KK11: algorithms for binary and linear searching

KK12: validation techniques, including existence checking, range checking and type checking

KK13: techniques for checking that modules meet design specifications, including trace tables and construction of test data

KK14: purposes and characteristics of internal documentation, including meaningful comments and syntax

Key skills

KS1: interpret solution requirements and designs to develop working modules

KS2: use a range of data types and data structures

KS3: use & justify appropriate processing features of a programming language to develop working modules

KS4: develop and apply suitable validation, testing and debugging techniques using appropriate test data

KS5: document the functioning of modules and the use of processing features through internal documentation

¹ All numbered dot points are from the VCE Applied Computing: Study Design 2020-2024, Copyright VCAA 2019 ([link](#))

Area of Study 2: Analysis and Design

- Learn and apply the Analysis and Design parts of the PSM to Software Solutions
- School-assessed Task (SAT) – Part 1 (Analysis & Design)
 - Create a project plan and Gantt chart
 - Analyse and Document a need or opportunity for a software solution
 - Generate design ideas and design a software solution

Key knowledge

Digital systems

KK1: security considerations influencing the design of solutions, including authentication & data protection

Data and information

KK2: techniques for collecting data to determine needs and requirements, including interviews, observation, reports and surveys

Approaches to problem solving

KK3: functional and non-functional requirements

KK4: constraints that influence solutions, including economic, legal, social, technical and usability

KK5: factors that determine the scope of solutions

KK6: features and purposes of software requirements specifications

KK7: tools and techniques for depicting the interfaces between solutions, users and networks, including use case diagrams created using Unified Modelling Language

KK8: features of context diagrams and data flow diagrams

KK9: techniques for generating design ideas

KK10: criteria for evaluating alternative design ideas and the efficiency and effectiveness of solutions

KK11: methods of expressing solution designs using data dictionaries, mock-ups, object descriptions and pseudocode

KK12: factors influencing the design of solutions, including affordance, interoperability, marketability, security and usability

KK13: characteristics of user experiences, including efficient and effective user interfaces

KK14: development model approaches, including agile, spiral and waterfall [NEW]

KK15: features of project management using Gantt charts, including the identification and sequencing of tasks, time allocation, dependencies, milestones and critical path

Interactions and impact

KK16: goals and objectives of organisations and information systems

KK17: key legal requirements relating to the ownership and privacy of data and information.

Key skills

KS1: select a range of methods to collect and interpret data for analysis

KS2: select and justify the use of an appropriate development model

KS3: apply analysis tools and techniques to determine solution requirements, constraints and scope

KS4: document an analysis as a software requirements specification

KS5: generate alternative design ideas

KS6: develop evaluation criteria to select and justify preferred designs

KS7: produce detailed designs using appropriate design methods and techniques

KS8: create, monitor and modify project plans using software.

Software Development Unit 4

This unit aims to deepen students' knowledge and understanding of the selected programming language through the continued development of their project that was started in Unit 3. They will also further develop their skills in validating data and testing code, ensuring that their solution functions as expected. Students will use their project plans to monitor progress and will evaluate both the quality of the solution and the effectiveness of this plan in monitoring their project. Students learn how to examine the security practices of an organisation and the risks to software and data during the development and use of the software solutions.

Area of Study 1: Development and Evaluation

- Learn more about programming and software development
- School-assessed Task (SAT) – Part 2 (Development & Evaluation)
 - Develop Software Solution and track progress against plan
 - Test and Evaluate the Software Solution
 - Write a Report on the Software Solution and Planning

Key knowledge

Digital systems

KK1: procedures and techniques for handling and managing files and data, including archiving, backing up, disposing of files and data and security

Data and information

KK2: ways in which storage media, transmission technologies and organisation of files affect access to data

KK3: uses of data structures to organise and manipulate data

Approaches to problem solving

KK4: processing features of a programming language, including classes, control structures, functions, instructions and methods

KK5: characteristics of efficient and effective solutions

KK6: techniques for checking that coded solutions meet design specifications, incl construction of test data

KK7: validation techniques, including existence checking, range checking and type checking [aka U3KK12]

KK8: techniques for testing the usability of solutions and forms of documenting test results

KK9: techniques for recording the progress of projects, including adjustments to tasks and timeframes, annotations and logs

KK10: factors that influence the effectiveness of development models

KK11: strategies for evaluating the efficiency & effectiveness of software solutions & assessing project plans

Key skills

KS1: monitor, modify and annotate project plans as necessary

KS2: propose and implement procedures for managing data and files

KS3: develop a software solution and write internal documentation

KS4: select and apply data validation and testing techniques, making any necessary modifications

KS5: prepare and conduct usability tests using appropriate techniques, capture results, and make any modifications to solutions

KS6: apply evaluation criteria to evaluate the efficiency and effectiveness of the software solution

KS7: evaluate the effectiveness of the selected development model

KS8: assess the effectiveness of the project plan in managing the project

Area of Study 2: Cybersecurity: Software Security

- Learn about the security risks to software and data during the software development and deployment
- Understand the legal and ethical issues facing software developers
- School-assessed Coursework (SAC)
 - Respond to a case study to examine and recommend security practices

Key knowledge

Digital Systems

- KK1:** physical and software security controls used to protect software development practices and to protect software and data, including version control, user authentication, encryption and software updates
- KK2:** software auditing and testing strategies to identify and minimise potential risks
- KK3:** types of software security and data security vulnerabilities, including data breaches, man-in-the-middle attacks and social engineering, and the strategies to protect against these
- KK4:** types of web application risks, including cross-site scripting and SQL injections
- KK5:** managing risks posed by software acquired from third parties

Data and information

- KK6:** characteristics of data that has integrity, including accuracy, authenticity, correctness, reasonableness, relevance and timeliness

Interactions and impact

- KK7:** reasons why individuals and organisations develop software, including meeting the goals and objectives of the organisation
- KK8:** key legislation that affects how organisations control the collection, storage (including cloud storage) and communication of data: the Copyright Act 1968, the Health Records Act 2001, the Privacy Act 1988 and the Privacy and Data Protection Act 2014
- KK9:** ethical issues arising during the software development process and the use of a software solution
- KK10:** criteria for evaluating the effectiveness of software development security strategies
- KK11:** the impact of ineffective security strategies on data integrity
- KK12:** risk management strategies to minimise security vulnerabilities to software development practices.

Key skills

- KS1:** analyse and discuss the current security controls to protect software development practices and to protect software and data
- KS2:** identify and discuss the potential risks to software and data security with the current security strategies
- KS3:** propose and apply criteria to evaluate the effectiveness of the current security practices
- KS4:** identify and discuss the possible legal and ethical consequences to an organisation for ineffective security practices
- KS5:** recommend and justify an effective risk management plan to improve current security practices.