Course Guide

SDV602 SOFTWARE DEVELOPMENT, 2ND LANGUAGE

(Level 6, 15 credits)

Te Pūkenga

Contents

Version History

Version	Effective date	Completed by	Reviewed by
1	July 2023	Todd Cochrane	Sandra Dyke
1.1	July 2024	Todd Cochrane	

Welcome to your course and to your Kaiako (tutor)

Nau mai, haere mai ki te akoranga hangarau matihiko
(Welcome to this Digital Technology paper)

Welcome this course - Nau mai ki tenei akoranga. Enjoy your time studying with Te Pūkenga, let's make the most of every opportunity that comes our way.

	Name	Contact details	Availability
Kaiako	Dr Todd Cochrane	Todd.Cochrane@nmit.ac.nz	During class times unless arranged via an email.
You may expect a response from your kaiako within two to three workdays. If you do not receive a response, please contact the coordinator or administrator below.			

This Course Guide contains essential course and assessment information about SDV602 SOFTWARE DEVELOPMENT, 2ND LANGUAGE. You will need to refer to this Guide regularly. Please discuss any parts of it that seem unclear with your Kaiako or Programme Support Coordinator.

Programme support contact

	Name	Contact details	Availability
Programme Support Coordinator	Camila Bernal	Email: camila.bernal@nmit.ac.nz Ph: 03 5463621 G Block room G127	Monday to Friday 8:00am to 4:00pm
If you do not receive a response, please contact our academic administration hub below			
Administrative contact	Lisa Salter	Email: lisa.salter@nmit.ac.nz Ph: 03 5395070 A Block room A131	Monday to Friday 9:00am to 5:00pm

Course, study, and support

Course and Study related information	Nelson Location	Description
Programme Regulation and Course Descriptors	Moodle – Information Technology Home Page	The programme regulations and course descriptors.
Orientation information	Moodle – Information Technology Home Page	General course related information including software requirements. General study skills information
Te Pūkenga Policies	Moodle – Information Technology Home Page	A link to all Te Pūkenga policies – a particularly useful one is Te Kawa Maiorooro.

As ākonga, you will also need information that is not specific to this course or your programme, like how to access the library and learning support, and what pastoral care is available. For this information, use the links below.

Service	Nelson Location	Contact Details
**Ākonga	Student Information** Guides and links for ākonga.	Website	**https://students.nmit.ac.nz/**
Library and Learning Support Services Get help with online research, using databases, academic writing, disability support, APA referencing, and any health and well-being needs.	M Block	learningsupport@nmit.ac.nz
Te Puna Manaaki A team of Māori support staff who help with academic, physical, spiritual and whānau assistance.	D Block	maorisupport@nmit.ac.nz
Pacific People Support Support for our Pacific students.	D Block	pasifika.support@nmit.ac.nz
SANITI All NMIT students are entitled to representation and services provided by the Association.	N Block	info@saniti.co.nz

Ehara taku toa i te toa takitahi, engari he toa takitini

(My strength is not that of a single warrior but that of many)

Course details

Course code and title	SDV602 SOFTWARE DEVELOPMENT, 2ND LANGUAGE		
Credits	15	**Learning hours total**	150
Pre-requisites	SDV503 Introduction to Software Development, or equivalent skills and knowledge		
Course aim	This course will broaden ākonga software development horizon by experiencing a new programming language and environment. By using a language, possibly from a different vendor and/or is aimed at a different hardware platform or environment ākonga will gain valuable and marketable expertise. Building on the prerequisite course(s), ākonga will apply the learnt analysis and design methodologies to the new programming environment, and if necessary adapt them to suit the characteristics of the chosen programming language.		
Learning outcomes	After you successfully complete this course, you will be able to…		
	1. Examine and show understanding of a new programming language and identify its purpose and characteristics.		
	2. Demonstrate independence in the investigation and effective application of language syntax features.		
	3. Effectively design and implement a software project in response to the requirements of a project brief. The software produced will be of an intermediate to advanced level.		
Link to online course site	SDV602 24NN-S2 Ecampus website		

Indicative learning hours	**Kaiako-directed**	60	**Work-integrated or Placement**	0	**Self-directed**	90
Kaiako-directed hours	You are expected to attend all kaiako-directed sessions on the timetable. You are more likely to succeed in the course if you attend regularly. Kaiako-directed sessions will include a range of learner centred activities, i.e., tutor seeded and guided (directed) learning of course theory and practice through individual and group work, guided team presentations, and practice tutorials, based on authentic situations.					
Self-directed hours	You are expected to spend at least six hours per week doing self-directed study for this course. This self-directed study supports and reflects your engagement with the course content. It is essential to your success in the course. You must use self-directed hours in this course to undertake project work and individual and group or team-based study, readings, and A/V tutorials.					

6

Resources Required

Required texts/readings	All the key resources including readings, video and other reference material and activities can be accessed through the online course. All course handouts, assessment details (including submission dropboxes) and any supporting material are available online. The online course is open to students for the duration of this course (i.e. Semester 2) See your course on the Te Pūkenga NMIT ecampus website, that contains proposed sessions and resources for each class.
Other course resources	You are expected to bring your own laptop PC that meets BIT BYOD specifications, see here. • Ākonga Study Essential • Academic Integrity Module • NMIT APA 7th edition referencing guides Research Skills: locating and evaluating information

Ka mate koe i te paoa; kāore, he kāuta

(A job that offers a good return or promotion will make up for some difficulties or inconvenience)

Schedule

The schedule below outlines how we will progress the course. The topics and timeline below are subject to change. Where possible, changes will be notified in advance. Please note sometimes the room allocated is subject to change. Please check your Moodle, or email and/or MyNMIT for changes to venues or topics.

Class Times

Day	Time	Room
Monday	15:00 – 17:00	A215
Wednesday	15:00 – 17:00	A215

Please note class times or rooms may change due to circumstances outside of Digital Technologies control. Please regularly refer to your timetable on your MyNMIT and Moodle posts from the Digital Technologies Team.

WEEK	Beginning	TOPIC	PRACTICE AND TUTORIAL
Week 1	22 July	Course outline and projects Milestone One setting the goals. How to compare programming languages Types: strong vs weak, dynamic vs static.	Work on LinkedIn tutorial (1) https://www.linkedin.com/learning/python-for-non-programmers/python-from-zero?autoplay=true&resume=false&u=76059146 https://replit.com/languages/python3 Alternative https://www.linkedin.com/learning/python-essential-training-14898805/getting-started-with-python?autoplay=true&u=76059146
Week 2	29 July	Data types Input, Print Flow control	What is good documentation? Guido's guides App structure … Work on LinkedIn tutorial (2) https://www.linkedin.com/learning/advanced-python/welcome?autoplay=true&u=76059146
Week 3	05 August	GUI, and Graphics: Charts Modules	GUI libraries – recommend/choose for course. https://www.linkedin.com/learning/rapid-application-development-with-python/rapid-development-of-python-gui-apps-with-wxglade?autoplay=true&u=76059146
Week 4	12 August	Command arguments Procedures and functions GLOBAL Input, output	

Week 5	19 August	List and or Dictionary Comprehensions	
Week 6	26 August	Milestone One review and work - Hand in Milestone 1 (30th of August 11:59 pm)	
Week 7	02 September	Milestone Two, setting the goals	Describe your Project 2 App's data and purpose.
Week 8	09 September	Map, Reduce Filter	https://docs.python.org/3/tutorial/datastructures.html
Week 9	16 September	Local Storage, database	to device file storage, To local database
Week 10	23 September	Milestone Two review and work - Hand in Milestone 2 (27th of September 11:59 pm)	
	30 September	Term Break	
	07 October	Term Break	
Week 11	14 October	Milestone Three, setting the goals	D
Week 12	21 October	Network services sockets and pipes.	Web Services / httpRequest
Week 13	28 October	Chat systems	Connecting to a networked db service No class on Monday (Labour day)
Week 14	06 November	Assessment week 1 Project refinement	
Week 15	11 November	Assessment week 2 Milestone Three review and work - Hand in Milestone 3 (15th of November 11:59 pm)	

No classes on the following dates:

Public Holiday
28th October Labour Day

He iti hau marangai e tū te pāhokahoka
(A little storm and then a rainbow appears)

Assessment Guide

Assessments For This Course

This course takes a project centred approach to work. The course learning outcomes and aims with their associated outcomes are integrated into the activities and assessments you undertake during completion of your project work.

You will meet requirements and skills in **Learning Outcome 3.** "Effectively design and implement a software project in response to the requirements of project briefs. The software produced will be of an "intermediate to advanced level" by undertaking projects as specified in the project briefs.

Learning Outcome 1. "Examine and show understanding of a new programming language and identify its purpose and characteristics."

Throughout the course as you develop your project your will gain knowledge and insight as follows:
a) The heritage and philosophy of the programming language is explored.
b) The platform(s) for developing and running software applications are analysed and understood.
c) The characteristics, strength and weaknesses of the new programming environment are understood.

Learning Outcome 2. "Demonstrate independence in the investigation and effective application of language syntax features...". In your practices as a software developer you are expected to develop and demonstrate how you developed your capacity and understanding as follows:
a) The syntax of the programming language is studied and applied to problems of intermediate difficulty.
b) The syntax differences between this language and an earlier one studied are understood, pitfalls and dangers of careless use are recognised.
c) The design and implementation of code libraries (for example using classes, modules, or precompiled code with headers) is explored and applied.

Throughout the course you will be working on *two* projects. Submission of you work is divided into three stages to be handed in at three milestones. At each milestone you hand in and present all the work you undertook towards that milestone. For the first two milestones you receive feedback on the work you submit at the milestone. The final milestone introduces new material; however, you are also required to submit all work you have undertaken in the project. At milestone three, additional marks will be given as specified for the third and final milestone, however no further feedback will be given.

Marks given at milestones one and three will be final (summative) marks that contribute to your overall mark for the project.

The project briefs with milestone marking schedules can be found at the end of this course outline.

In addition to working on your projects you are expected to participate and attend the course. You are to collect evidence of your participation and practise as your work on course, the best way to do this is to set up an online storage location (GitHUB repo) that can be shared with your tutor to store material you produce as you study and work on SDV602.

Assessment Schedule (due dates may change)

Assessment	What	Due Date		Weighting
Project	Two Projects in three milestones (two assignments) as follows	See course schedule		
Assignment 1 (40%)	Hand in at Milestone 1 (summative)	(Week 6 of the course) 30th of August 11:59 pm		40%
Assignment 2 (60%)	Part 1 Hand in at Milestone 2 (indicative)	(Week 9 of the course) 27th of September 11:59 pm		20%
	Part 2 Hand in at Milestone 3 – The Final Hand in (summative)	(Week 15 of the course) 15th of November 11:59 pm		40%
		Total		100%

Assignments will only be accepted if handed in on or before the due date unless there are special circumstances that are discussed with the course coordinator prior to the due date.

Criteria to pass this course:

- Genuinely attempt all assessments
- Meet all learning outcomes
- Gain a course result of C (50%) or higher

E kore te matua e rawe ki te moana takai ai, engari anō a uta

(Don't wait until the last minute to do something, be prepared)

SDV602 PORTFOLIO PROJECT 1

Assessment: Assignment 1

Course: SDV602 SOFTWARE DEVELOPMENT, 2ND LANGUAGE

Given: 22nd of July 2024

Due   ~~30th of August 11.59 pm, 2023~~

Weighting: 40% of the course grade

Tutor: Todd Cochrane

Moderator: _____

This assignment focuses on the following course learning outcomes.

1	Examine and show understanding of a new programming language and identify its purpose and characteristics.
2	Demonstrate independence in the investigation and effective application of language syntax features.
3	Effectively design and implement a software project in response to the requirements of a project brief. The software produced will be of an intermediate to advanced level.

CONDITIONS

- For this assessment you are to write your own writing in a journal. Hand in material as specified on your class website, in the course dropbox.
- APA 7TH in-text citations and references are required for any sources.
- The assessment may be submitted to Turnitin.
- The assessment is to be done individually.
- This In this assessment, you may NOT use generative artificial intelligence (AI) to generate materials or content in relation to work undertaken during the assessment. Marks will only be given for work that expresses your capacity in the subject matter, see following the condition.
- All work submitted for assessment must be entirely your own.

PORTFOLIO PROJECT 1 BRIEF

Starting with the scene-to-scene game presented in class with source code from the class Ecampus website. Replace the current text comparison command interface with a command parsing system with actions that implement:

- ~~Movement from one scene in the game to another. (Current implementation replaced)~~
- ~~Fighting with monsters.~~
- ~~Picking up inventory.~~
- Status for example: "score" that returns the player's score and lists their inventory.

Modules (or Classes) to be implemented for your project.

1. **Command Parser** "command_parser.py"
   Implement the command parsing system as a separate module or class that is imported into the main GUI system.
2. **Monster Fight** "monster_fight.py"
   ~~Implement a monster fighting system as a separate module or class that is imported into module "command_parser.py".~~
3. **Inventory** "inventory.py"
   Implement an inventory system as a separate module or class that is imported into the modules "command_parser.py" and ~~"monster_fight.py".~~
4. **Status** "status.py"
   Implement a status system as a separate module or class that is imported into the module "command_parser.py"

PROJECT 1 (MILESTONE ONE) DETAILS: MARKING SHEET

DUE DATE: as specified in the course outline.

TOTAL MARKS: 120 marks (worth 40% of the course)

Part 1 PROJECT MODULE IMPLEMENTATION (LO2 and LO3) 100 marks

Using the App provided in class and in your course Moodle website, implement and integrate the four modules described above into a single Python application. Requirements are as follows, and as given in the rubrics below:

- The App must have its own virtual environment set up (venv) that includes all the public sourced modules used by your App.
- The code is to be uploaded into a GitHUB repository that includes a "Read Me" document that indicates how to run the App.
- The completed App is to show how to use Python data types: Tuple, List, Dictionary and Set, and make use of comprehensions.
- Each module is to be set up with an "if main" section.
- The modules are to be organised into their own folder or folders and set up so that they can be used by the Python module system.

The rubric for implementing each of the four modules and integrating them into the App is as follows (80 marks).

0	1-4	5-8	9-14	15-20
Not submitted	A named module is present. Some functionality is implemented	A named module is present in a suitable folder and the Python module system can find it. Some functionality and is implemented. But it does not run from the App.	A named module is present in a suitable folder and the Python module system can find it. Most functionality is implemented. The module is integrated and runs from the App. The code is commented, but not following the Python PEP guide. One or more of the data types are used.	A named module is present in a suitable folder and the Python module system can find it. All functionality is implemented. The module is integrated and runs from the App. The code is commented, following the Python PEP guide. One or more of the data types are used and described.

The rubric for adjusting the App to include the new command parser and to work in a virtual environment. (20 marks)

0	1-4	5-8	9-14	15-20
Not submitted	The App has been adjusted to call the parser. However, it does not operate.	A virtual environment has been created and public modules have been installed in the virtual environment. The	A virtual environment has been created and public modules have been installed in the virtual environment. The App has been adjusted to call the parser. Most functionality	A virtual environment has been created and public modules have been installed in the virtual environment. The App has been adjusted to call the parser. Most functionality

	There is no virtual environment.	App has been adjusted to call the parser. Some functionality is present.	is present. The code is commented, but not following the Python PEP guide. One or more of the data types are used.	is present. The code is commented, following the Python PEP guide. One or more of the data types are used and described.

One mark is deducted for each error identified when the App runs.

<mark>Part 2 PROJECT REFLECTIVE REVIEW VIDEO (LO1) 20 Marks</mark>

Create a five-minute video that meets the following requirements.

Work on the course is based on two projects through which you demonstrate your understanding and capacity to undertake web development as covered in the course. A reflective review is a record of your reflection on your practice as you learn about another programming system.

Marking of the Reviews

They must reflect what you have learnt and developed as you became familiar with the new programming system. An overall grade will be awarded according to the following rubric. The marker will ask "have all of the sections been covered? Are the entries reflecting on the practices learnt and covered?". **A score out of 20** will be awarded as the grade for this assessment.

The best approach in this course is prepare for your review by writing a short reflective note each week on the topics you are studying to cover the learning outcome to "Examine *and show understanding of a new programming language and identify its purpose and characteristics.*"
In your notes compare the new programming language with the one you have already studied.
For each programming system, the new and the one you have experienced, write to cover the following.

a) <mark>The heritage and philosophy of the programming language is explored</mark>
b) <mark>The platform(s) for developing and running software applications are analysed and understood.</mark>
c) <mark>The characteristics, strength and weaknesses of the new programming environment are understood</mark>.

Marking rubric

0	1-8	9-14	15-17	18-20
No submission	There is no discussion on type systems. The programming system is described in	One or more of the following is missing: The	Some minor aspects are missing from the following:	The programming system is described in

terms of its a) heritage and philosophy. b) platforms and code libraries c) The characteristics, strength and weaknesses	programming system is described in terms of its a) heritage and philosophy. b) platforms and code libraries c) The characteristics, strength and weaknesses – including a description of type system	The programming system is described in terms of its a) heritage and philosophy. b) platforms and code libraries c) The characteristics, strength and weaknesses – including a description of type system	terms of its a) heritage and philosophy. b) platforms and code libraries c) The characteristics, strength and weaknesses – including a description of type system with examples

How to submit your project work.

Hand in all source code and any documentation in one folder that you have uploaded to a GIT repository. Using your course website put the URL of your GIT repository into the dropbox provided for this milestone. <mark>Make sure you name your repository OR a folder in your repository SDV602 Milestone 1 <your student name>.</mark>

SDV602 PORTFOLIO PROJECT 2

Assessment: Assignment 2 Data Explorer Screens

Course: SDV602 SOFTWARE DEVELOPMENT, 2ND LANGUAGE

Given: At time specified in the Course Outline

Due   In two parts:
27th of September 11:59 pm
15th of November 11:59 pm

Weighting: 60% of the course grade

Tutor: Todd Cochrane

Moderator: _____

This assignment focuses on the following course learning outcomes:

1	Examine and show understanding of a new programming language and identify its purpose and characteristics.
2	Demonstrate independence in the investigation and effective application of language syntax features.
3	Effectively design and implement a software project in response to the requirements of a project brief. The software produced will be of an intermediate to advanced level.

CONDITIONS

- For this assessment you are to write your own website solutions, and reports. Hand in material as specified for each milestone as described below and on your class website, in the course drop box.
- APA 7TH in-text citations and references are required for any sources.
- The assessment may be submitted to Turnitin.
- This assessment is to be done individually.
- In this assessment, you may NOT use generative artificial intelligence (AI) to generate materials or content in relation to work undertaken during the assessment. Marks will only be given for work that expresses your capacity in the subject matter, see following the condition.
- All work submitted for assessment must be entirely your own.

PORTFOLIO PROJECT 2 BRIEF

In this project you demonstrate your understanding of the programming language and capacity to undertake analysis and development in that programming system, as covered in the course.

You are to develop an application that provides live and interactive graphical displays, that include charts, to enable presentation of different graphical views of data sources for a business (or scientific) analyst. Each live interactive graphical display allows exploration of data from a

data source – these are <mark>Data Explorer Screens (DES)</mark>, see figure 1. <mark>At each DES the app accepts text input and mouse clicks</mark> and <mark>displays a brief description</mark> of the situation that relates to the information being displayed in an adjustable chart (you are to use the code libraries discussed in class). <mark>The application provides data explorer screens.</mark> Each screen includes a chart and accepts text-based input, as well as pointer (i.e. mouse based) input. Each DES presents views of data in a chart displayed as well as a summary of data fields.
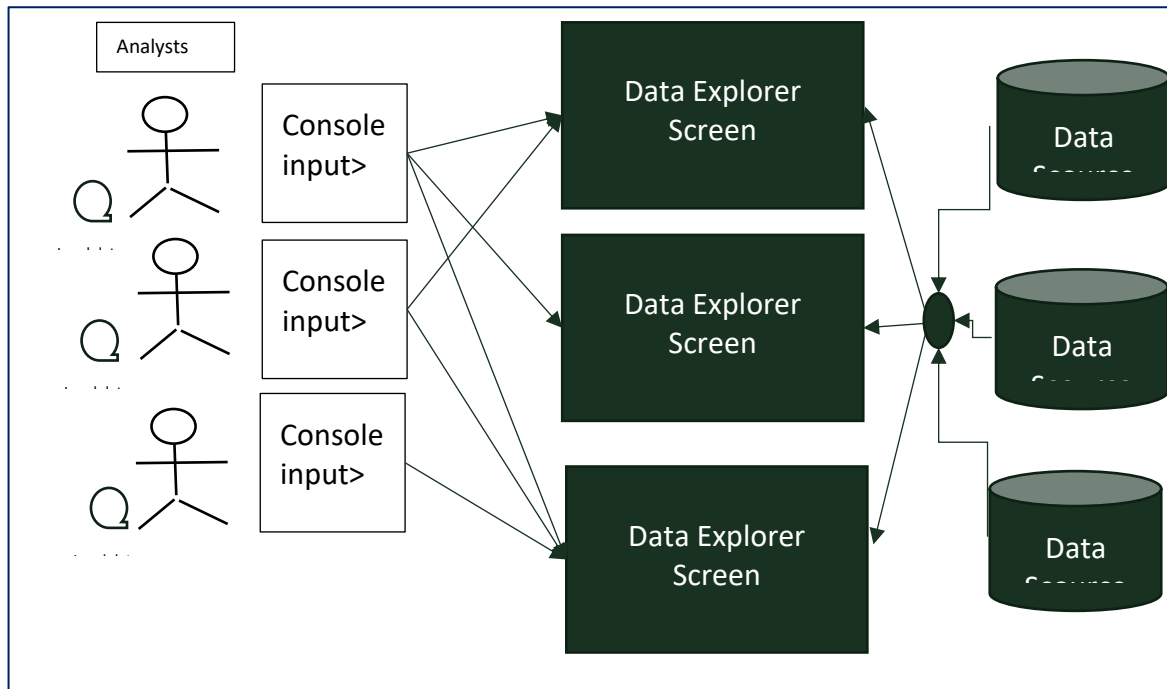

Figure 1 Application sketch, see figure 2 for DES details. Each analyst starts with a console input.

The application is to provide console text-based input to control and display at least three different DESs, see figure 2. Each DES presents charts (each based on their individual data source) and a summary or comparative analytical information. The final system is to provide a chat system for analysts viewing the current DES, that is accessed using a remote http-based API through a remote data service. The final system is to retrieve data from a shared remote data source. The screens are to include a number of options for views of the data at each display.
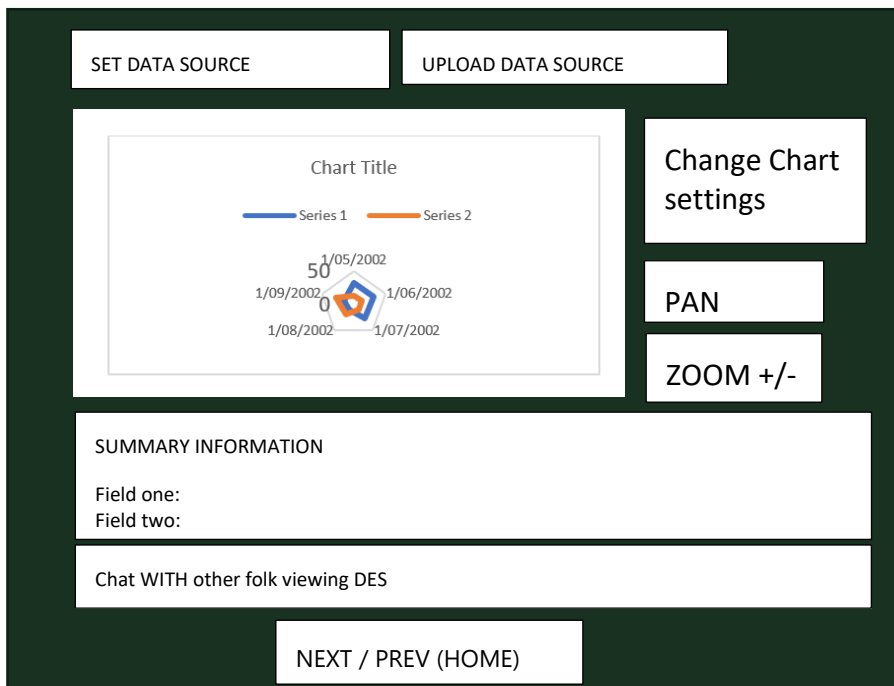
Figure 2 Data Explorer Screen - indicative sketch.

The FINAL application is to allow more than one analyst to view the same DES at the same time and upload and retrieve data on a shared remote server. Two or more analysts can communicate with each other using a text chat system (that is to be included as part of each DES) when they are viewing the same screen. The final application is to include user login and logout.

Technical overview

You are required to create a set of storyboards that that make up your implementation of the supporting software. The system you develop is to be an implementation of your design. A set of test scripts should be developed as part of your system developments. You will be influenced by the architecture determined by the software development platform, however a suitably identified architecture beyond this is preferred in the development of your application.

Required from you.

You are to write a one-page description of your application, its purpose, and a brief overview of the parts of the application. Send that description to your tutor by email, the day before the first class of the second week.

Assignment 2 is in two parts as follows. Part 1 is to be handed in at Milestone Two and Part 2 is to be handed in at Milestone Three.

SDV602 Assignment 2 part 1. (MILESTONE TWO) DETAILS: MARKING SHEET

DUE DATE: as specified in the course outline.

TOTAL MARKS: 60 marks (worth 20% of the course)

Brief - In Summary

Create a storyboard depicting ALL the screens and interaction in the application. Include a description of your app and narrative descriptions of the interactions depicted on each storyboard.

Create a first prototype of your app that implements prototypes for three interactive DESs for the application (without user login and logout).

Work detail and Marking schedule.

The application needs to have three DESs and navigation between all.

1. App Design and Documentation (18 marks) LO 2,3

You are to hand in:

A brief description of the purpose of your application. (6 marks)

0	1-2	3-4	5-6
No description	Brief description without details for screens	Description with Screens detailed	Description with Screens detailed. Motivation for the application (business or scientific)

A set of storyboards depicting the working of the WHOLE application, including for example Login and Chat systems. Each storyboard is to include a table that details ALL interactions, inputs and outputs. (12 marks)

Marks are deducted for missing aspects.	Storyboard The storyboards depict a screen	Details (2) All actions and displays are labelled and described, must include how to move from one screen to another	
Login			_/3
DES One			_/3
DES Two			_/3
DES Three			_/3
		Total	_/12

2. *Implement tests scripts that runs three DESs* (36 marks) LO 2, 3

Six marks per screen.

Each DES:

- accepts correct input and displays correct information (6 marks)
- displays a place holder for the chart (6 marks)
- provides navigation to other DESs, must include a top command interface to display them all (6 marks)

The screens must run without presenting errors.

For each DES additive marks	Correct input, (2) Display (2)	Placeholder (2) produces a Chart (2)	Navigation has a command console interface to run DESs , (2) Moves to a different screen (2)	Total
One				_/12
Two				_/12
Three				_/12
Total				_/36

3. *Coding practices (up to 6 marks*) LO 2 – partly must look up programming language standard and guides (e.g., in Python the PEPs) to get this right.

- Comments using the convention of the programming system (2 marks)

  Reason for mark:

- Code formatting according to established programming system conventions, (2 marks)

  Reason for mark:

- Naming (constants, variables, procedures, classes, modules) according to established programming system conventions (2 marks)

  Reason for mark:

How to submit your project work.

Hand in all source code and any documentation in one folder that you have uploaded to a GIT repository. Using your course website put the URL of your GIT repository into the dropbox provided for this milestone. Make sure you name your repository OR a folder in your repository SDV602 Milestone 2 <your student name>.

SDV602 Assignment 2 part 2 (MILESTONE THREE) DETAILS: MARKING SHEET

DUE DATE: as specified in the course outline.

TOTAL MARKS: 120 (worth 40% of the course)

Milestone Three has three sections.

SECTION 1 ARCHITECTURE AND LOCAL DATA STORAGE AND RETRIEVAL (LO 2,3) 30 marks

Brief - In Summary

Starting with the previous three DES implementations, complete local data reading, uploading and merging, that is immediately displayed for each DES. Build application architecture using suitable abstractions. (In Python, use Modules, and possibly Classes, think about "domain specific language style" architecture). Implement "local" upload.

Work details and Marking schedule.

Application Architecture
The application uses a Python module that supports code reuse for the three DESs screens. Each DES reuses the module (15 marks)

For each DES additive marks	Module is used by Each DES (3)	Each DES makes call to all domain methods or procedures in the Module (6)	The call to each domain method on the module produces a change in each DES (6)	Total __/15
Total				__/15

Local Data Storage
Data for DES is read from the local disk (Use **local data services)** (15 marks)

For each DES additive marks	Each DES displays a different graph based on the data source (3)	Each DES chart display can be adjusted, for example "Zoom in"," Zoom out", "Pan". (3)	Each DES includes an upload function, that uploads a suitable dataset for its graph display (4)	Each DES displays new data merged with existing data after upload (5)	Total __/15
Total					/15

Is operating without errors **One mark deducted for each error.**

SECTION 2 Implement Multi-user aspects of your DES App. (LO 2,3) 60 marks

Brief - In Summary

Implement, multi-user aspects of your App. Compare Python with another programming language.

Work details and Marking schedule

The marking will look carefully at the way the system is implemented and check that the system handed in is operating without errors.

1. *Implement multi-user aspects of your app* **(30 marks)**

   Include:
   a. A log in facility. (12 marks) - divided into
      i. ~~Validate the login using a remote database system. (6 marks)~~
      ii. Maintain and manage session status in an encapsulated manner. (6 marks)
   b. Create a chat system, that displays the analysts viewing the same screen. (18 marks) divided into:
      i. Works and is tested to completion (6 marks)
      ii. Clear use of architecture and separation of concerns (6 marks)
      iii. Clear App structure (6 marks)

2. Implement remote storage are retrieval of data sources.
   Data for DES is read from a remote data store (Use **remote data services) (30 marks)**

For each DES additive marks	DES displays a different graph based on the remote data source (10)	DES includes an upload function, that uploads a suitable dataset to the remote data store for its graph display (10)	DES displays new data merged with existing data after upload, from the remote data store (10)	Total __/30
Total				

Is operating without errors **One mark deducted for each error.**

Examine and show understanding of a new programming language and identify its purpose and characteristics.

Write a short at least 700-word **essay** as follows. **(30 marks)**

The essay is to present the programming language or system:

a) The heritage and philosophy of the programming language is explored (10 marks)

b) The platform(s) for developing and running software applications for that language are analysed and described. (code libraries, IDEs) (10 marks)

c) The characteristics, strengths and weaknesses of the programming environment are described. (10 marks)

Marks are given according to the following rubric.

	0	1-4	5-7	8-10
a) The heritage and philosophy of the programming language is explored	Not covered	Heritage: the story of when, how and why the programming language was established. Without details or citations. Philosophy: Presents anecdotal perspective of the culture, and how that is maintained. Missing details	Heritage: the story of when, how and why the programming language was established. Without citations. Philosophy: Presents anecdotal perspective of the culture, and how that is maintained.	Heritage: the story of when, how and why the programming language was established with a brief biography of key players and their contribution – this may include quality committees. Philosophy: describes the paradigms applied in its development, and how that becomes a culture and continues to be maintained. Includes APA7 citations

b) The platform(s) for developing and running software applications for that language are analysed and described. (code libraries, IDEs)	Not covered	Parts of the following are missing. Presents one platform and describes how it works. Describes how the code libraries are used.	Presents one platform, and describes how it works. Describes how the code libraries are used.	Presents a scope for the platforms, and describes them. Describes how the code libraries are used, with APA7 citations.
c) The characteristics, strengths and weaknesses of the programming environment are described.		Parts of the following are missing. The programming language type system is described with examples. Relates advantages and disadvantages of the system overall.	The programming language type system is described with examples. Relates advantages and disadvantages of the system overall. Misses type system details and or advantages or disadvantages	The programming language type system is described with examples. Relates advantages and disadvantages of the system overall. Include APA7 citations

Note APA 7TH in-text citations and references are required for any sources.

How to submit your project work.

Hand in all source code and any documentation in one folder that you have uploaded to a GIT repository. Using your course website put the URL of your GIT repository into the dropbox provided for this milestone. Make sure you name your repository OR a folder in your repository SDV602 Milestone3 <your student name>.

Assessment regulations and special circumstances

If you have any questions about doing the assessments for this course discuss these with your kaiako.   The following regulatory information is provided to help you – if you need assistance with regulations. If you cannot find the information that you need, please email details to your kaiako.

Assessment regulations for this course
See your Programme Regulations and course descriptors for assessment regulations.

The following links contain important information about policies and procedures specific to this course. Please look at these and discuss any questions with your kaiako.

Te Kawa Maiorooro – Educational Regulatory Framework

Part 2: Enrolments and withdrawals – including timelines for withdrawals.

Part 3: Recognition of prior knowledge and skills.

Part 5: Learning, Teaching, and Training – including participation, engagement, progress, and standards of behaviour.

Part 7: Assessment – including extension policy and late/non-submission of assessment, academic integrity, and reconsideration options.

Te Pūkenga Policies -link to all policies

Ākonga Appeals Policy

Ākonga Concerns and Complaints Policy

Use of Generative AI

The ability to use generative AI will be set out in each assessment. Check carefully if and how this can be used. If you do use generative AI, make sure you keep a copy of the results and how you have developed your ideas. An easy way to do this is to print the page with the results and prompts used and save this to your own files. There are many tools to print webpages, a useful one is GoFullPage which can be included as a browser extension.

Read Artificial Intelligence (AI) – Guidelines for Ākonga (students) to find out more about:

- Acceptable and unacceptable use.
- Using, acknowledging, and referencing AI.

Special Assessment Circumstances	Details (e.g. provisions, rationale, procedures, restrictions, penalties)
Resits or Resubmissions	Ākonga who do not pass a summative assessment may request one resit or resubmission of that assessment provided they genuinely attempted and submitted the assessment by the due date and time (if an extension was granted this becomes the due date). Any resit or resubmission is undertaken according to the following: • the ākonga requests permission from the Curriculum Area Manager (or delegate) within 5 working days of the return of the marked assessment; • permission is granted by Curriculum Area Manager (or delegate); • any approved resit or resubmission is completed within 5 working days. If a resit or resubmission is granted, ākonga will be advised which one of the following is required: • resit or resubmission of the whole assessment, or • completion of another assessment which measures the same outcomes, or • resit or resubmission of the part of the assessment that did not meet pass criteria. The maximum mark available for any resit or resubmission of an assessment or part of an assessment is 50%.
Extensions	Work submitted for summative assessment will have an assigned due date. Work to be assessed must be submitted on or before the time designated by the tutor on the due date to qualify for marking, unless an extension has been granted. Ākonga requiring an extension beyond the due date should follow the following process: • advise the subject tutor or course facilitator that an extension is required before the due date, not after. • advise the tutor or course facilitator of the reason for the extension, and will be required to provide evidence of this. Approval is at the tutors or course facilitator's discretion. Extensions beyond the course end date must be approved by the Curriculum Area Manager
Late assessment submission	Ākonga may choose to submit an assessment after the due date up to, and including, five days past the due date (without seeking a formal extension). However, 10% of the mark will be deducted per day up to a maximum of 50%. Any deduction for late submission will be indicated on the marking schedule. Assessments received more than five days after the due date will not be marked.

Alternative Assessment Arrangements	If ākonga is unable to attend a summative assessment for genuine reasons beyond the control of ākonga and/or extenuating circumstances, ākonga should first advise the course tutor. On recommendation from the course tutor, an application may then be made to the Programme Coordinator to sit the assessment at a different time or place. Wherever possible as much notice as possible should be given to the Programme Coordinator in order to give time to make alternative arrangements.
Aegrotat Pass	If ākonga performance in a summative assessment is affected by factors beyond the control of ākonga, ākonga may apply to the Curriculum Area Manager for consideration for special assessment. Examples include sickness, injury or bereavement. • Aegrotat consideration is only available for up to 50% of a course's total assessment • The minimum achievement of 50% in course work indicates eligibility for an aegrotat application, not an indication that the application will be successful. (Other criteria will be used to assess an aegrotat application.) • All decisions will be at the discretion of the Curriculum Area Manager Consideration for an aegrotat will be restricted to a maximum of 25% of the total credits for the qualification.
Compassionate Consideration	If ākonga performance in a summative assessment is affected by factors beyond the control of ākonga, ākonga may apply to the Programme Coordinator for compassionate consideration. Possible options may include alternative assessment arrangements (see above) *Note: Re-sits and resubmissions carried out under compassionate consideration conditions have no limit on the mark available.*
Reader/Writer Assistance or other special assistance for a summative assessment	Ākonga wishing to receive special assistance in order to undertake a summative assessment shall apply in writing to the Administration Office no later than three weeks prior to the date of the assessment. Such applications must state the nature of the disability and the type of assistance required.
Conceded Pass	May be awarded at the discretion of the Academic Standards and Quality Committee.
Reconsiderations (Re-marks)	Ākonga may seek reconsideration of any assessment by applying in writing to the Programme Coordinator, within 10 working days of receiving the result, setting out the grounds for reconsideration.
Marks Carried Forward	Not available
Other Assessment Regulations	Not applicable
Assessment in Te Reo Māori	Request must be registered prior to the start of the course, and approval is required from the Curriculum Area Manager

Assessment Results

- Individual assessments may cover one or more of the learning outcomes.
- The result for each assessment is given as a percentage mark.
- Each summative assessment is assigned a percentage weighting.

Course Results

The overall percentage mark for the course is calculated by adding the weighted results for all summative assessments. To derive the course result the overall percentage mark is converted into a grade using Course Result Key AC-NMIT-06. Refer to the Course Descriptor for details of the requirements for successful completion of each course.

AC-NMIT-06:

RESULT	MARK RANGE (%)	DESCRIPTION
A+	85 - 100	
A	80 - 84	Pass with Distinction
A-	75 - 79	
B+	70 - 74	
B	65 - 69	Pass with Merit
B-	60 - 64	
C+	55 - 59	
C	50 - 54	Pass
D	40 - 49	
E	0 - 39	Fail grades

Other results that may be awarded:

RECORDED RESULT	DESCRIPTION
CC	Pass - Course credited on basis of <u>similar</u> course already completed as part of another approved qualification from NMIT or other institution
CT	Pass - Course credited on basis of <u>same</u> course already completed in another qualification at another institution
RPL	Pass - Recognition of Prior Learning
Grade (AEG)	Aegrotat Pass - Where a result is awarded following consideration of special circumstances. A grade is able to be determined.
Pass (AEG)	Aegrotat Pass - Where a result is awarded following consideration of special circumstances. A grade is not able to be determined.
D (CON)*	Conceded Pass - Where a narrow fail in a course is compensated by good grades in other courses within the programme.

W	Withdrawn
DNC	**Did Not Complete – where a compulsory element of a course is not submitted, or the ākonga formally withdrew after 80% or informally withdrew at any time.**

Note: A conceded pass may be awarded at the discretion of the Academic Committee if ākonga has fulfilled the minimum course requirements but failed narrowly to achieve the standard over all assessments, or completed most but not all of the required work at an acceptable level. The Academic Committee must be satisfied that ākonga has worked conscientiously and is worthy of special consideration.

Ākonga is eligible to receive only one conceded pass in any one year in the same programme. A conceded pass is not available on a second result.

Attendance/Engagement

TECHNOLOGY ENHANCED DELIVERIES	
Attendance/participation requirement	International ākonga on campus must comply with Immigration NZ attendance requirements. Absences will be reported to immigration New Zealand and may adversely affect Visa status.
Attendance/participation recommendation	Ākonga are more likely to succeed if they maintain regular attendance. They are advised of this in the Programme Outline/Handbook and/or by their tutor. Failure to attend all scheduled/supervised learning and teaching sessions may also adversely affect the eligibility to receive loans and/or allowances for domestic (NZ) ākonga. Attendance at scheduled class sessions will be recorded centrally using the NMIT Student Management System, and registers can be accessed by all staff.
Process for reporting absences/non-participation	Ākonga are responsible for notifying the Curriculum Area Administrator or class tutor of any absence, by the first scheduled class session of the first day of absence. The Administrator is responsible for recording receipt of such notification in ākonga file, and for advising the tutor(s) of ākonga absence.